DOCUMENT RESUME

ED 131 987 95 RC 009 569

AUTHOR	Stromsdorfer, Ernst W.; Moayed-Dadkhah, Kamran
TITLE	A Pilot Cost-Effectiveness Analysis of the Economic and Educational Impact of the Mountain-Plains Education and Economic Development Program, Inc. A Research Study for Department of Health, Education and Welfare, National Institute of Education. Revised May 1976.
INSTITUTION	Indiana Univ., Bloomington. Dept. of Economics.
SPONS AGENCY	National Inst. of Education (DHEW), Washington, D.C.
PUB DATE	May 76
CONTRACT	NIE-C-74-102
NOTE	252p.; Some pages in the appendix may not reproduce well
EDRS PRICE	MF-$0.83 HC-$14.05 Plus Postage.
DESCRIPTORS	*Career Education; *Cost Effectiveness; *Economically Disadvantaged; *Evaluation Methods; Human Capital; Models; Productivity; *Program Evaluation; *Rural Population
IDENTIFIERS	*Mountain Plains Program

ABSTRACT

Presenting a cost-benefit analysis of the Mountain-Plains Career Education Program (a family based program for the economically deprived in the mountain plains states operating out of Glasgow Air Force Base in Montana) and the methodological basis for a full and more detailed study, this evaluation includes: (1) discussion of theoretical issues that must be considered in any cost-effectiveness or cost benefit analysis, with special reference to the project under study; (2) a general framework designed to analyze different aspects of the accumulation of human capital (trainee employability and productivity as measured by wage rates); (3) a model of short-run cost including a discussion of the program's data deficiencies, possible remedies, and some program specific conceptual problems, as well as an estimation of the program cost function; and (4) an econometric appendix designed for future investigations based upon more solid methodological ground. In general, this assessment indicates that the results of the evaluation are distorted due to "extremely poor" data reporting and an ill chosen comparison group. Eleven detailed proposals for data collection are presented as suggestions for future data collection procedures. (JC)

Documents acquired by ERIC include many informal unpublished materials not available from other sources. ERIC makes every effort to obtain the best copy available. Nevertheless, items of marginal reproducibility are often encountered and this affects the quality of the microfiche and hardcopy reproductions ERIC makes available via the ERIC Document Reproduction Service (EDRS). EDRS is not responsible for the quality of the original document. Reproductions supplied by EDRS are the best that can be made from the original.

rec 15-11-76

SCOPE OF INTEREST NOTICE
The ERIC Facility has assigned this document for processing to: RC
In our judgement, this document is also of interest to the clearinghouses noted to the right. Indexing should reflect their special points of view. CE

ED131987

Department of Economics
Indiana University
Bloomington, Indiana

A Pilot Cost-Effectiveness Analysis of the Economic and Educational Impact of the Mountain-Plains Education and Economic Development Program, Inc.

A Research Study for
Department of Health, Education and Welfare
National Institute of Education

by

Ernst W. Stromsdorfer
U. S. Department of Labor and
Department of Economics, Indiana University

and

Kamran Moayed-Dadkhah
Department of Economics
Indiana University

Revised May 1976

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

This report was prepared for the Department of Health, Education and Welfare/National Institute of Education under Contract USOE NIE C-74-102. Since contractors conducting research and development projects under government sponsorship are encouraged to express their own judgment freely, this report does not necessarily represent the official opinion or policy of the National Institute of Education or the U.S. Department of Labor. The authors alone are responsible for the content, analysis and interpretation of the data reported herein.

CE009569

PREFACE

This report is a pilot analysis of the economic and educational impact of the Mountain-Plains Education and Economic Development Program, Inc. It was designed to ascertain if a full-fledged cost-benefit evaluation of the program was feasible and valid. In no sense are the results of the present study to be taken as a definitive impact and cost analysis. Rather, this study is intended to point out data and design strengths and weaknesses of the Mountain-Plains effort.

It has, unfortunately, pointed out a wider array of methodological weaknesses than of strengths in the design and execution of the experiment. Perhaps the political and social history of this interesting project made such a result inevitable. Nevertheless, the final judgment on the economic and educational effectiveness of Mountain-Plains remains to be stated.

Ernst W. Stromsdorfer
Kamran Moayed-Dadkhah

ACKNOWLEDGMENTS

We would like to thank Dr. Michael Hock, project officer of this study, who supervised it from beginning to end. Also thanks are due to the administrators of the Mountain-Plains Education and Economic Development Program, Inc., who provided us with the necessary data, especially Mr. David Coyle, Mr. Gary Smith and Mr. Luther Robison.

Different versions of the report were typed with great care and skill by Mrs. Tina Keller and Mrs. Joan Hongen. We would also like to thank Mrs. Linda Parker and her staff in the Department of Economics, Indiana University, who provided us with secretarial help. Mrs. Barbara Barnes of the U.S. Department of Labor typed the final revision. Wrubel Computing Center's facilities were used for the computations and development of the programs at the end of this report. Mr. Robert Hogan performed all the necessary computer programming.

Ernst W. Stromsdorfer
Kamran Moayed-Dadkhah

TABLE OF CONTENTS

SUBJECT PAGE

CHAPTER ONE -- INTRODUCTION.................... 1

The Objective of Cost-Benefit Analysis.... 3

The Necessity of Cost-Benefit Analysis.... 4

The Meaning of the Cost Function.......... 8

On Defining Cost Functions................ 10

The Definition of Output.................. 15

Treatment of Capital Cost................. 17

Earnings Foregone by Trainees............. 19

Estimation of the Benefits and the Concept of the Comparison Group......... 23

CHAPTER TWO -- BENEFITS........................ 25

The Model................................. 26

Estimation of Benefits to the Family vs. Estimation of Benefits to the Individual 39

Empirical Estimates of the Benefits of the Program.............................. 45

Conclusion................................ 54

SUBJECT PAGE

CHAPTER THREE -- COST.......................... 62

Types of Cost.......................... 66

Cost Function.......................... 88

Total Government Cost.................. 101

Conclusion............................. 102

APPENDIX A -- Econometric Issues............... 104

APPENDIX B -- Input Deck to Search Program "UBLSQ"......................... 123

APPENDIX C -- Program Description for Error Component Model................ 124

LIST OF TABLES

CHAPTER TWO PAGE

1. The Effect of Program Participation on the Labor Market Performance of Head of the Household.......................... 49

2. The Effect of Program Participation on the Labor Market Performance of the Spouse.................................. 50

3. The Effect of Program Participation on Total Family Income........................ 51

CHAPTER THREE

1. N.I.E. Payments to the Department of Defense, in Dollars........................ 72

2. Transfer Payments to Trainees, in Dollars 74

3. Total and Average Recruitment/Follow-Up Cost (Monthly Data), in Dollars............ 78

4. Total and Average Recruitment/Follow-Up Cost (Moving Average), in Dollars.......... 79

5. Trainee Relocation Costs (Monthly Data), in Dollars................................ 83

6. Trainee Relocation Costs (Moving Averages), in Dollars................................ 84

7. Variable and Total Costs (Monthly Data), in Dollars................................ 86

PAGE

8. Total and Average Total Costs (Moving Averages), in Dollars........................ 89

9. Estimates of the Coefficients of the Cost Function.................................... 96

LIST OF FIGURES

PAGE

CHAPTER ONE

Figure 1 -- Optimal Level of Output........... 7

CHAPTER TWO

Figure 1 -- Hypothetical Earnings Path of Trainees and Controls............. 38

CHAPTER THREE

Figure 1 -- Average Transfer Payments to Trainees........................... 75

Figure 2 -- Average Relocation Cost (Moving Average Data)..................... 85

Figure 3 -- Average Total Costs (Moving Average Data)..................... 87

CHAPTER ONE

INTRODUCTION

This is a cost-effectiveness study of the Mountain-Plains family based career education program. As such it has a two-fold aim: First to perform a cost-benefit study of the program and second, by means of the study design, methodology and analytical techniques, to lay the groundwork for a full and more detailed study of this program. Thus, we describe the actual methodology and theory used in this study as well as the potential methodology and techniques that could have been used had the necessary data been available. We note at the outset that the major handicap in this study was the lack of data, a problem which we will discuss further below.

This study is divided into four main parts. In the Introduction we discuss essential theoretical issues that must be considered in any cost-effectiveness or cost-benefit analysis, with special reference to the project under study. Chapter Two addresses the question of program benefits. In this chapter a general framework is developed for analyzing different aspects of the accumulation of human capital by trainees. Human capital is not an observable variable, therefore we consider the change in observable variables such as employability and productivity of the trainees (as measured by their wage rates) as a reflection of acquired human capital. The model can be used to estimate the opportunity cost of trainees' time but lack of data prevents us from empirically estimating this opportunity cost. Chapter Three is concerned with the estimation of the program cost function. A model of short run cost is discussed and estimated. The deficiencies of data, possible remedies, and some conceptual problems specific to this study are discussed.

There is an econometric appendix to this study. This appendix is not a comprehensive treatment of the subject. It serves as a reminder to future investigators what can be done if the intention is to base such a study as this on more solid methodological ground. In the present study we were not able to utilize some of the methods discussed due to the lack of appropriate data. But with the identification of the problems we note and an effort on the part of the administrators of the program, acquisition of the necessary data should not be a problem for future researchers. Use of some of these techniques requires extensive programming and econometric skill. We have provided the programs used in this study. A printout of the computer program, the necessary control cards and instruction on how to interpret the output is provided.

The Objective of Cost-Benefit Analysis

A manpower program aims at improving the lot of relatively less fortunate citizens in society.

In practice, a manpower program is a production process in that several inputs are brought together to produce an output. The inputs are instructors' services, trainees' time and so on. Output is the skill or acquired human capital the trainees have acquired as a result of going through such processes.

In this process, the value of the output should justify the use of the inputs, in the sense that the value that society puts on the output should be at least equal to the amount spent on the inputs. The reason for this requirement is that the value of inputs reflects their desirabilities in alternative uses. If society sacrifices those alternative uses then surely it should receive in return at least as much.

The purpose of cost-benefit analysis is to determine the relevant values to compare. What are the costs and what are the benefits of a program? Who gains the benefits? Who pays the cost? In fact this economic analysis goes further; it tries to determine not only if a program is worth its cost but also among several programs, which is the best and what is the optimal level of operation of the desired program.

The Necessity of Cost-Benefit Analysis

Let us assume that we have a measurable and conceptually accurate index of output, then we can define a cost function and a benefit function, relating on one hand the cost of the program and on the other hand the benefit of the program to the program output. The essential problems of cost-benefit analysis are to formulate and estimate the appropriate cost and benefit functions. The problem is conceptually straightforward and can be carried out with an analogy to the economic theory of the firm as follows:

Let

$C = C\ (q)$ cost function

$B = B\ (q)$ benefit function

where

C = cost

B = benefit

q = output

Then for any level of output q^o the benefits are such that the cost is justified if

$$b = B - C = B\ (q) - C\ (q) \geq 0$$

The optimum level of output is where the excess of benefit over cost is maximized, thus,

$$\frac{db}{dq} = \frac{d}{dq}\ (B - C) = B'\ (q) - C'\ (q) = 0$$

or $B'(q) = C'(q)$

where $B'(q) = \frac{d\,B(q)}{dq}$ and $C'(q) = \frac{d\,C(q)}{dq}$

gives the necessary condition for the optimal level of output. This condition implies that the optimal level of output is reached when marginal benefit (the difference between the benefit of two successive levels of output) is equal to the marginal cost (the difference between the cost of the same two successive levels of output).

Necessary and sufficient conditions for the optimal level of output are:

(1) $\frac{db}{dq} = B'(q) - C'(q) = 0$, and,

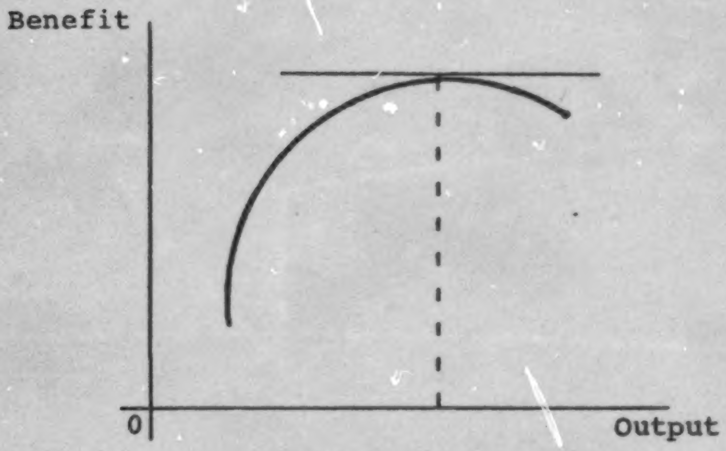
(2) $\frac{d^2b}{dq^2} = \frac{d\,B'(q)}{dq} - \frac{d\,C'(q)}{dq} = B''(q) - C''(q) \leq 0$

where $B''(q) = \frac{d\,B'(q)}{dq}$ and $C''(q) = \frac{d\,C'(q)}{dq}$

That is, if we plot the difference between benefit and cost for different levels of output against the level of output in a diagram, the resulting curve should be convex (concave to the origin), as in Figure 1

Figure 1

Optimal Level of Output



The condition for the desirability of the expansion of a program is[1]

$$\frac{db}{dq} = B'(q) - C'(q) > 0$$

and for the contraction of a program to be desirable

$$\frac{db}{dq} = B'(q) - C'(q) < 0$$

---

[1] Unless there are multiple maxima for $b = B(q) - C(q)$ in which case a small movement in one direction may not be optimal but a large movement will be a definite improvement.

The Meaning of the Cost Function

The above discussion demonstrates that a manpower program is a process through which certain inputs are utilized to produce a desired output. The resources (inputs) used up in the process have some price whether they are paid for explicitly or not, and therefore the process costs society, and may cost the government and individuals involved, depending on how the program is financed. To make any decision regarding a program or a group of programs one has to know the relationship between cost and output.

Going back to our crude outline of the process depicted above, we see that we have a set of expenditures and a set of outputs; thus, mathematically speaking, the cost function is a transformation relating the two sets. But this scheme is too general to suit any practical purpose. We have to make our cost function more specific by utilizing some limiting assumptions. Following the neo-classical theory of the firm, we make the assumption that only those combinations of inputs are used that result in a minimum level of cost for any given level of output.[2]

Now we have a unique relationship[3] between cost and output; this is the function we want to estimate. However, we still have a few important problems to resolve, namely, what should be defined and measured as output and the treatment of capital cost, to which we turn, after defining the cost function in a more rigorous manner.

---

[2] We can qualify this assumption by stating that the cost is minimized given the information available to the management of the program. In other words cost is minimized when the cost of gathering further information on how to reduce the cost is included in the total cost.

There is no argument with the fact that these are not satisfactory behavioral assumptions. Better alternatives may be forthcoming as the result of recent developments in the economic theory of bureaucracy where more realistic assumptions such as that of the budget maximizing organization are utilized.

[3] Note that if we qualify the cost minimization assumption by the availability of information, a change in the availability of information will shift the cost function. More important, the continuation of a program should result in learning by doing and therefore gradual reduction of cost, ceteris paribus.

On Defining Cost Functions

In economics we speak of costs and cost functions. But our definition of cost is somewhat different from what in financial data is termed cost (and which economists typically call expenditure). In order to understand what is meant by a cost function we have to understand the nature of the production function or program process. By production function we mean a process which relates different combinations of inputs to different levels of output. In mathematical notation we have

(3) $q = f(x_1, x_2)$

where

q = output

x_1, x_2 = inputs

The number of inputs is not necessarily confined to two; we chose two for simplicity of exposition. One can, if one chooses, think of x_1 as labor and of x_2 as capital. The most important feature of a production function is that it is assumed that it represents technical efficiency. That is, if by using a given level and combination of x_1 and x_2 we can produce different amounts of q, the production function relates that level and combination of inputs to the highest amount of output that can be produced. In other words, the basic assumption is that the technology used in the production process is the one yielding the highest possible output from any input mix.

On the other hand we know that the cost of producing a given amount of output consists of the sum of expenditures on inputs plus a fixed amount of cost which goes for setting up the production process. Naturally enough, the sum of expenditures on inputs varies directly with the level of output, while the fixed cost has to be incurred irrespective of the amount of the output. Thus, denoting the price of the first and second input by r_1 and r_2, respectively, fixed cost by b and total cost by TC, we can write

$$(4) \quad TC = b + r_1x_1 + r_2x_2$$

Now, we define the total cost function as the relationship between the sum of fixed cost and expenditures made on inputs used in the production of an output and that output, provided the choice of input combination is made so that the least expenditure for a given level of output is achieved. We may further elaborate this discussion by noting that one can change the input mix and still get the same output. This is because inputs are substitutable; we can increase one input to compensate for the decrease in another. Thus, rational behavior compels us to choose that input mix, among all input mixes giving the same output, which costs the least.

In sum, we have a production function, equation (3), and a total cost function, equation (4), both of which are linked to each other through the assumption of rational decision making. We can put these in a concise mathematical form as follows:

Minimize $TC = b + r_1x_1 = r_2x_2$

Subject to q^o as a fixed level of output.

It can be shown mathematically that the solution to this problem is equivalent to the solution of the following problem:

$$\text{Minimize } L = b + r_1 x_1 + r_2 x_2 - \lambda(q^o - f(x_1, x_2))$$

where λ is the Lagrangian multiplier and all other terms are defined as above. Necessary conditions for L to be minimum are[4]

$$(5) \quad \frac{\partial L}{\partial x_1} = r_1 + \lambda f_1 = 0$$

$$(6) \quad \frac{\partial L}{\partial x_2} = r_2 + \lambda f_2 = 0$$

$$(7) \quad \frac{\partial L}{\partial \lambda} = q^o - f(x_1, x_2) = 0$$

where $f_i = \frac{\partial f}{\partial x_i} \quad i = 1, 2$

Dividing equation (6) into equation (5) we get

$$(8) \quad \frac{r_1}{r_2} = \frac{f_1}{f_2}$$

---

[4] Here we are assuming that second order (sufficient) conditions are fulfilled.

Condition (6) above together with the production function, equation (1) forms a system of two equations and three unknowns, x_1, x_2, and q. We can solve for two variables in terms of the third one. Thus we can write:

$$x_1 = g_1(q)$$

$$x_2 = g_2(q)$$

inserting in equation (4) above we get:

$$(9) \quad TC = b + r_1 g_1 (q) + r_2 g_2 (q);$$

since r_1 and r_2 are constants, we can write equation (9) as

$$(10) \quad TC = b + c(q)$$

This is the total cost function. If we are interested only in variable cost, i.e., that portion of cost which varies directly with output, we can omit b and we will have

$$(11) \quad VC = c(q)$$

where VC denotes variable cost. On the other hand if we wish to know how much, on the average, a unit of output costs (either in total or variable cost terms), then we divide both sides of equations (10) and (11) by q to get:

$$\text{Average total cost} = \frac{TC}{q} = \frac{b + c(q)}{q} \text{ and}$$

$$\text{Average variable cost} = \frac{VC}{q} = \frac{c(q)}{q}$$

$$\text{Marginal total cost} = \text{Marginal variable cost} =$$

$$\frac{d}{dq}\, c(q) = c'(q)$$

The Definition of Output

The output of a manpower program in the form of human capital acquired by trainees as a result of the program is not an observable variable. We can, however, measure the impact of this additional human capital in the labor market in the following forms:

1. The change in the real wage rate of the trainees,
2. The change in the trainees' prospect of employment, and
3. The change in the wages and employment of non-trained labor (the negative effect that this trained manpower will have on other workers).

The problem with an index based on these changes is that it will depend on the value of output[5] (price times quantity) and we want a measure of output in terms of quantity only. An alternative is to ask how many people received the output of the program in the form of training and with what intensity? Then the weighted sum of individuals receiving the training by, say, the grades they received, will form the output index we are looking for. Here, dropouts will cause a problem. It is likely that dropouts receive partial training which will help them later on but we have no indication of how they performed in the program. If the percentage of dropouts is small and they closely resemble the average person of the target population and were not drawn to the program by misinformation, then we can consider completors and dropouts as the joint output of the program. Unfortunately, in the present study we do not have information on the completor-dropout status of the participants in a way that can be linked to the cost of the program and we do not have information on the intensity of training.

---

[5] These labor market effects are what we call pecuniary benefits of the program as distinct from real effects, i.e., effects which result in a net increase in welfare for society.

Also, due to the lack of data we defined output in the cost chapter differently from output in the benefit chapter. In the cost chapter we used family-trained-per-month as our measure of output and in the benefit chapter, time spent in the program is the proxy for output. Neither is satisfactory, and more important, because of this difference, a comparison of costs and benefits is highly biased. What aggravates the problem is that on the cost side our unit of observation is the family and on the benefit side it is individual trainee. This is a most important point and efforts have to be concentrated on production of more reliable and meaningful output data.

Treatment of Capital Cost

In a manpower program a part of expenditures is spent on durable assets such as buildings, machinery, equipment and so on. The difference between these expenses and those on instructors' salaries or utilities is that while the expenses are made in one or several installments, their services to the program are extended over a period of time. If we treat these expenses the same way we treat current expenses, we will distort the relationship between cost and output. The period in which the expense is made will show an unduly high average cost and in other periods we will have (again unduly) low average cost.

One device to overcome this difficulty is to prorate such costs over the period of the useful life of the capital equipment in question. We use the following formula

$$CCS = \frac{Co\ r\ (1+r)^n}{(1+r)^n - 1}$$

where

CCS = cost of capital services

Co = total capital expenditures

r = rate of interest

n = useful life of capital

In this study the total amount of expenditures on equipment was so low compared to the total of all expenditures that such a proration would not have made any difference in our results and, therefore, it was not attempted.

Earnings Foregone by Trainees

It was mentioned above that one of the inputs into the manpower program is the time of the trainees. Had the trainee not entered the program, there is a probability that he would have gotten a job and therefore contribute to his own as well as to social output. Joining the program deprives the society of this output. Similarly, from his standpoint, the trainee foregoes the earnings he would have got had he not participated in the program and the government loses the tax it could have collected on that income.

There are different views on how to estimate these foregone earnings. If there were full employment in the society, there would be no problem at all because once a group of people is removed from the labor market, the full amount of their earnings, reflecting their contribution to the total output, could have been counted as foregone earnings. Unfortunately, less than full employment is the normal state of the world, in particular for persons manpower programs are aimed at. The question is, when some trainees might have been unemployed had they not joined the program, what is the appropriate imputation of foregone earnings?

There are two polar views on the subject. First, there are those who argue that in the presence of unemployment, when the labor of trainees is withdrawn from the market, other people replace them immediately and there is no loss of output to society and no loss of taxes to government. Only individual trainees will lose the equivalent of the earnings they would have earned had they not trained. There are, on the other hand, some economists who argue that any resource should be valued at the value of its marginal product and this is the wage rate paid to his peers. The fact that there is unemployment should not change the argument; the failure to use a resource should not affect its value. Our position is in between these two and much closer to the latter. Suppose there is unemployment among the instructors from which we hire the teachers for a program. Once they are hired they are paid the full wage. What is, then, the difference between this wage rate and wages imputed for rational economic calculations? We believe none. But there is one point to consider; if we take the full earnings of the trainees as a part of social cost, then two projects equal in all respects except for different unemployment rates among their target populations will have the same cost function. This will distort the results. To overcome such bias we recommend the use of the expected opportunity cost notion, which is the amount of earnings foregone times the probability of getting a job (as measured by one minus the unemployment rate).

It is interesting to note that those who join the program will not necessarily lose all the income they would have earned. Thus, what we have to do is to compute the change in the expected earnings of the trainees as a part of the cost of the program. Unfortunately in the case of the present study because of the lack of complete data on trainees and controls and significantly, because of the lack of data on the labor market characteristics of trainees during their participation in the program, we can only take the data on controls as the opportunity cost for trainees. The reason behind this is that we have to assume the trainees would not join the labor force, would not become employed and would receive as income only their allowance from the program. Thus, whatever the controls received was what the trainees would have received had they not participated in the program. Subsistence allowances and other payments trainees receive from the program are transfer payments from government and the difference between what controls earn and this transfer payment (which can be positive, zero or negative) is the individual opporunity cost of the trainees.

Estimation of the Benefits and the Concept of the Comparison Group

As mentioned above, the components of benefit we will estimate are the effects of participation in the program on labor force participation, employability and the wage rate. These effects of the program should be measured by comparing the difference of these variables for the case in which the trainee joins the program and the case in which he does not. But it is clear that a person cannot be in and out of the program at the same time. Labor force participation, employment and the wage rate of trainees before joining the program cannot be assumed to represent their status had they stayed out of the program because the general trend of the economy will exert considerable effect on both their employment and earnings. This leads us to the concept of the comparison group. By comparison group we mean a group of the people whose socio-demographic characteristics are as close to those of trainees as possible. Thus the difference between labor market characteristics of the two groups after the completion of the program by trainees will reflect the change, over different time intervals, of these characteristics attributable to the time spent in the program.

If the comparison group's socio-demographic characteristics exactly matched those of trainees, then we must compare the following ratios between trainees and the comparison group: The ratio of participation in the labor force to the total number of each group; the ratio of employed people to the total number of labor force participants in each group; and, the average wage rate of all those employed in each group.

This concludes our discussion of the basic concepts. Now we turn to the study of benefits and costs of the program. In each respective chapter a model is developed for estimating benefits and costs of the program, the data is evaluated, the results of the estimation as well as the conclusions are presented and finally, the direction for further data collection and future research is discussed.

CHAPTER TWO

BENEFITS

This chapter addresses itself to the question of measurement of the benefits of the program. By benefits we mean changes in the desired labor market characteristics of trainees that can be attributed to the program. Thus we have several issues here: First, we have to identify the desired labor market characteristics; second we must measure these changes; third, we have to factor out the influences of sources other than program participation; and, finally we must specify a measure of program participation where different levels of changes in labor market characteristics of trainees can be attributed to the change in this program variable.

The chapter is organized in the following way. First we set up a model by which we can analyze our problem and estimate the effects indicated above. Then the elements of this abstract theory are identified with the variables for which we have data from the Mountain-Plains project. Then we discuss the econometric problems we have and how we estimated the benefit equations. Finally the results of the estimation are presented and discussed.

The Model

Previously we talked about the manpower program as a process of production. In this process of production the trainees start with a certain set of characteristics; socio-demographic, economic and so on. We divide this set into two subsets. The first subset, denoted by X, includes all socio-demographic characteristics of the trainee such as sex, age, race, education and his economic characteristics prior to joining the program such as his employment status, wage rate and so on. In short, X consists of all those characteristics that cannot be changed as a result of program participation. Thus let us denote the second set by B, which includes all the labor characteristics of the trainee, a change in which constitutes the benefits to the trainee, and, by extension, to society. Explicitly, B consists of three elements: Labor market participation, employment status and the wage rate.

Labor market participation. A person who is not in the labor force, that is, a person who is neither employed nor is seeking work, behaves so for several reasons. It may be the case that he is not of working age (children and retired persons), that he is unable to work (handicapped), that he need not work (because of parent or spouse support or wealth), that he does not know any trade to offer in the market, that he thinks he won't be able to secure employment because he has been unemployed for a long time, or, finally, that he prefers to live as an appendage to the society on welfare or illegal means rather than by full labor force participation.

In each of the last three reasons above a change from non-participation in the labor force to participation may constitute an improvement for the individual. In all cases the change signifies an important change of attitude of the trainee towards himself and society. Thus such a change can mean a benefit to the individual and a benefit to society.

Employment status. The acquisition of skill that improves one's employability should be considered as a benefit to the individual trainee. This change signifies the fact that the trainee no longer belongs to that group of "not-needed" people. In the parlance of economics it is no longer true that there is no demand for his services.

If we consider the output produced by this newly employed individual, the society is better off because more output exists. Government is better off because it pays out less welfare payments to the trainee and receives more taxes from him. However, there remains a disturbing problem in that the newly employed trainee may have replaced another worker who is now unemployed. If such has happened there will be no change in the total number employed and no necessary change in total output of the society.

This is one of several secondary effects of a manpower program which, like the problem of externalities in general, is much referred to but never rigorously investigated.

The problem of externality is an important qualification to the second component of our benefit index. However, its effect should not be exaggerated. In some studies after the problem of externalities and difficulties surrounding their quantification is mentioned, the analysts often take the liberty to interpret this vagueness and indeterminancy of results in the direction which suits their purpose. The fact is that even if the existence of external effects is ascertained, we are not sure of their magnitude; they may be negligible or of immense effect. On net, they can be positive or negative.

In the absence of an operational concept of externality and the lack of data we can do nothing but fall back on the employability measure, always keeping in the back of our mind the qualification of externalities.

It should be noted that although we talked of employment status as a binary variable, that is, a person is either employed or unemployed, this need not be the case. Employment status is in fact a continuous variable. Given a time period T, during this period a person can be employed for any portion of it say α , where

$$\alpha = \frac{t}{T} \text{ where}$$

$$0 \leq t \leq T$$

and t is the period during which the person has been employed. Now α can vary from zero to one. α is the employment status of the person. In some studies ~~employment~~ status is defined as a dichotomous variable in which case α is defined in the following way:

$$\alpha = \begin{cases} 0, \text{ if } t = 0 \\ 1, \text{ otherwise} \end{cases}$$

The reason for this specification usually lies in the way data is collected. In this study we defined α as the average number of weeks employed on all jobs when all employment is converted into a full time equivalent. Thus our α is a continuous variable.

Wage rate. The wage rate is the compensation for the use of a unit of labor. As such it measures the contribution of the marginal person to the value of output. But in simple terms, the wage rate measures the value to the employer of the work of the last person hired in a firm. Since at some point the contribution of labor to output decreases as the number of employees increases, the contribution of the last person hired to output is smallest. It follows that when an employer has to pay a uniform wage rate to all employees, he will be better off if he pays the least possible; namely the value of the product of the marginal person. To put this more rigorously, take a profit maximizing firm when technology and the amount of capital is fixed and therefore output depends on the amount of labor utilized; the profit of the firm will be:

$$\begin{aligned} TT &= R - C \\ &= P \cdot F - b - WL \end{aligned}$$

where

TT = total profit
R = total revenue
C = total cost
P = price of output
F = the production function
L = labor
b = fixed cost
W = wage rate

Note that because technology and the amount of capital is fixed the only variable cost is wages and salaries. The necessary condition for profit maximization in this case is:

$$\frac{dTT}{dL} = PF' - W = 0$$

or $W = PF'$

where $F' = \frac{dF}{dL}$

Thus the employer equates the value of the marginal product of labor to the wage rate.

In this sense the wage rate, when the labor market is in equilibrium is a measure of the marginal productivity of labor.[1]

---

[1] When the labor market is not in equilibrium, the wage rate will be indeterminate; where its upper limit is the value of marginal product and its lower limit will be the marginal value of leisure to the worker. In any case the wage rate is a good approximation to the value of the marginal product of labor.

Thus when a trainee goes through the program, one measure of whether he has learned a useful skill is his post-program wage rate. If everything else is equal, and his wage rate is the same as the typical person in the comparison group who hasn't gone through the training process, then one might question whether the program has contributed to his skill.

With this introduction it becomes clear that the wage rate is an indicator of the effect of the program on a trainee's economic performance. And the change in the wage rate that can be attributed to the program can be counted, definitely, as a benefit of the program. In the present study we had to use the average earnings of trainees as a benefit measure because of the way the data is collected on this program. Earnings is defined as

$$E = W \cdot N$$

where

W = wage rate

N = amount of time worked

Thus, in this measure the amount of time worked enters the picture and, therefore, we can not use earnings as a measure of the value of the marginal productivity of labor. However, we do not have any other option but to use the earnings measure.

Having discussed the two sets of variables--the set of socio-demographic characteristics, X, and labor market characteristics, B--we now turn to their relationship. As it is clear B depends on general economic behavior and if general economic conditions change with time, then B is going to fluctuate over time. Also, in any given economic situation the elements in B, i.e., labor force participation, employment status and the wage rate (or earnings) of trainees (and the comparison group), depend on the socio-demographic characteristics of each individual. Thus, we can write B as a function of X, that is,

$$B = F(X)$$

note that B and X are vectors and this is a functional relationship between two vectors and therefore can be written as:

$$B_i = F_i\ (X_1, X_2, X_3, \ldots, X_n) \quad i = 1, 2, 3, \ldots, n.$$

where B_i is the i-th element of vector B and the X's are elements of X. Assuming a linear relationship we can write

$$B = AX$$

and for the i-th element of the vector B for the j-th individual, for example, for the employment status of the tenth person, we can write

$$B_i^{\,j} = A_i\ X^j$$

where A_i is the i-th row of matrix A and X^j is the set of socio-demographic characteristics of the j-th person.

Now we can incorporate the effect of participation in the program and length of stay in the program on labor market performance of trainees in the following way. Let S^j denote the length of stay in the program by the j-th trainee, then if the S affects labor force participation, employment status or wage rate (or earnings) of trainees, we can write:

$$B_i^{\,j} = A_i\ X^j + \lambda_i\ S^j$$

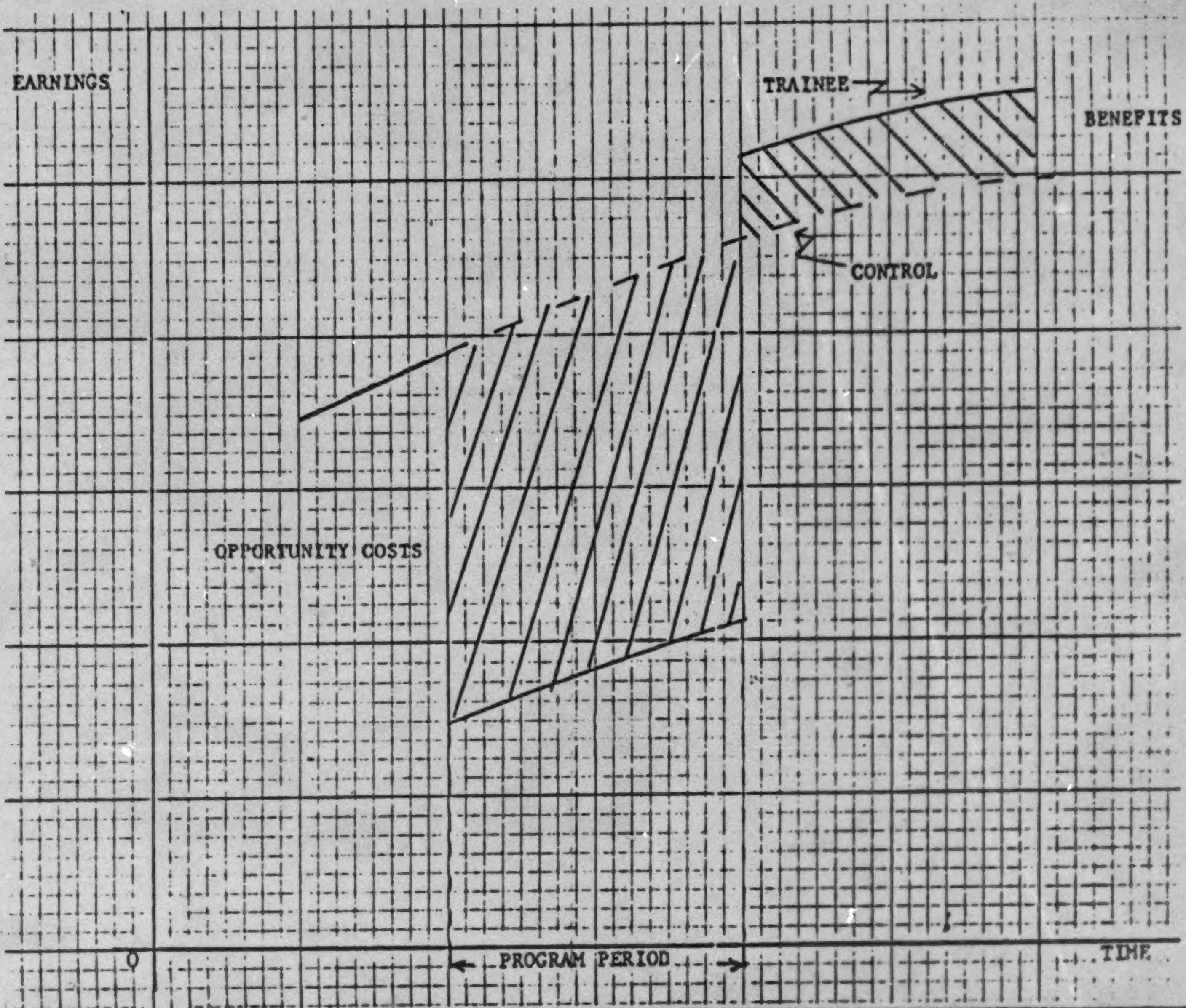
needless to say for the comparison group $S^j = 0$.

To estimate the effect of the program on each component of B we need to estimate the λi's. Each of these λ's shows the average effect of the length of stay in the program on the labor market performance of trainees as opposed to the comparison group members who have zero length of stay in the program.

Figure 1 depicts the earnings of trainees and comparison group before, during and after participation in the program. As can be seen, Figure 1 is drawn on the assumption that trainees and the comparison group come from the same population. That is, on average, they will have the same lifetime earnings profile in the absence of the program. When trainees join the program their earnings, compared to the control group's, will drop. It may go down to zero in the case that trainees are not allowed to accept any employment during their training period or, (as is depicted in Figure 1) they may still earn something. In any case the difference between the earnings of the two groups measures the opportunity cost to the trainees. After the completion of the program, if in fact the trainee has acquired new skills, his earnings will be greater than the controls who did not go through this same training. Hence over time the earnings of trainees will stay above the earnings of the controls and the difference will be the benefit attributable to the program. The same area between the two curves will also measure the benefit to the society if the secondary and external effects as discussed above can be assumed away.

This is the general description of our model for estimating the benefits of the Mountain-Plains program. Needless to say earnings was singled out for the purpose of illustration. All the variables which form the components of the B vector can be considered in turn. It should be noted, also, that in drawing Figure 1 we assumed the lifetime path of the variables under consideration for trainees and controls to be parallel. However this need not be assumed so. In the present study, because of lack of data, we could not estimate the effect of time on the benefits or costs of the program. In other words in the figure

FIGURE 1 - HYPOTHETICAL EARNINGS PATH OF TRAINEES AND CONTROLS



we will assume that participation in the program both during and after the completion of training affects the intercept (height) but not the slope of the lifetime path of labor force participation, employment status and earnings of the trainees. It is clear that this is a restrictive assumption and one should allow for the change in costs and benefits over time.

Before turning to the application of our model to the actual program under study and presenting our empirical results we should discuss an important problem regarding the interaction between labor market performance of the families in which both husband and wife participate in the program.

Estimation of Benefits to the Family vs. Estimation of Benefits to the Individual

The families who participated in the Mountain-Plains program were of two types: 1) Families in which both husband and wife participated in the program and, 2) families in which only the head of the household participated. The economic decisions of these two types of families differ from each other in many respects and these differences will be ultimately reflected in the labor market performance of each. Without trying to depict a formal economic theory of the household and the effect of both husband and wife working, we would like to give a brief explanation of the differences between the two types of families.

A family consisting of husband and wife and one or more children has a given amount of time (say 16 hours per day for each husband and wife) to devote to earning income, performing household labor and leisure. Given the wage rate and economic conditions in general and given the relative skills of husband and wife, the family has to make a decision in allocating its time between these three activities. This is done on the basis of a family's preferences. If the wage rate for the wife is such that it is lower than her reservation wage rate (that is, the wage rate below which she prefers to stay out of the labor market), the whole burden of acquiring income will be on the husband. This dictates certain restrictions on his working time to meet the preference of the family for income. On the other hand if both husband and wife work, it is the total amount of income that counts, given, of course, the increased opportunity costs of both holding a job. This also dictates certain limitations on the place of work and, perhaps, even the type of work, for both of them. These two factors, namely the income of the wife and the job location restriction, will affect the labor force participation, employment and earnings of the wife and husband. Thus going back to our main scheme where

$$B_i^j = A_i X^j + \lambda_i S^j$$

we can now write a system of equations; one equation for the husband and one for the wife

$$B_i^{jh} = A_i X^{jh} + \lambda_i S^{jh} \quad \text{husband}$$

$$B_i^{jw} = A_i X^{jw} + \lambda_i S^{jw} \quad \text{wife}$$

Now if B_i^{jw} has some effects on B_i^{jh} and vice versa we should write:

$$B_i^{jh} = A_i X^{jh} + \lambda_i S^{jh} + \gamma_i^h B_i^{jw}$$

$$B_i^{jw} = A_i X^{jw} + \lambda_i S^{jw} + \gamma_i^w B_i^{jh}$$

This is a system of simultaneous equations. Now if we estimate $B_i{}^{jh}$ and $B_i{}^{jw}$ without any consideration of this problem of simultaneity we will have biased our results. In other words we have overlooked information which was important for the estimation of the λi's. Note that we can estimate each equation separately or solve the system for $B_i{}^{jh}$ and $B_i{}^{jw}$ and estimate the reduced form, in which case the system will be:

$$B_i^{jh} = A_i X^{jh} + \lambda_i S^{jh} + \gamma_i^h (A_i X^{jw} + \lambda_i S^{jw} + \gamma_i^w B_i^{jh})$$

$$B_i^{jh} = \frac{1}{1- \gamma_i^h \gamma_i^w} (A_i X^{jh} + \lambda_i S^{jh} + \gamma_i^h A_i X^{jw} + \gamma_i^h \lambda_i S^{jw})$$

similarly,

$$B_i^{jw} = \frac{1}{1- \gamma_i^h \gamma_i^w} (A_i X^{jw} + \lambda_i S^{jw} + \gamma_i^w A_i X^{jh} + \gamma_i^w \lambda_i S^{jh})$$

Observe that the socio-demographic characteristics as well as the participation of both husband and wife have some effect on the labor force performance of the other. Note that the second set of equations can be estimated using ordinary least squares in which case we do not get the estimate of λi's, rather we estimate $\frac{1}{1-\gamma_i^h \gamma_i^w} \lambda_i$; that is, the effect of the program on the family after interactions between the labor force performance of husband and wife has been accounted for. If we desire an estimate of the λ_i, and at the same time take the simultaneity of the two equations into consideration we have to use one of several techniques dealing with the system of simultaneous equations. These techniques are two-stage least squares, limited information-maximum likelihood, three stage least squares and, full information maximum likelihood. These techniques are more sophisticated than ordinary least squares but at the same time they are more costly and more vulnerable to the deficiency of the data as well as to misspecification error. For example, if only one equation is misspecified or the pertinent data for one equation is unreliable, if we use OLS only that particular equation is affected, provided the problem of simultaneity does not exist. On the other hand if we use three stage least squares, then the estimates of all equations will be affected and the misspecification will creep into the parameter estimates of all equations.

The data we have is anything but reliable or complete. Moreover our small number of observations do not allow us to use all the variables we deem necessary as our explanatory set. Therefore we cannot rely on these more sophisticated estimation techniques. Furthermore, it is clear that if we consider the interaction between husband and wife, the number of our independent variables will double. With our sample size already being very small, the inclusion of these variables will leave us with no degrees of freedom. As a result we have had to assume away this problem of simultaneity. However, to the extent that it was possible, we incorporated the effect of the labor force performance of couples on each other, as will become clear in the reporting of our empirical results.

Having set the stage we now turn to the presentation of our empirical results.

Empirical Estimates of the Benefits of the Program

In order to estimate the benefits of the program in terms of labor force participation, employment and earnings for the head of the household and spouse we chose the following variables reported by Mountain-Plains trainees.

1. Average number of weeks employed, head of household (HOH)
2. Average monthly income, (HOH)
3. Present employment status, (HOH)
4. Average number of weeks worked, spouse
5. Average monthly income, spouse
6. Present employment status, spouse
7. Total family income

For the measures reflecting the effect of the program we used the following program status variables:

1. HOH completed the program, regardless of the status of spouse
2. Spouse completed the program, regardless of the status of HOH
3. Comparison group member

Thus we are measuring the difference between participation of the completors and dropouts and the difference between comparison group members and dropouts.

It is clear now that we cannot measure the change in the labor force participation of trainees, also, we cannot measure the effect of length of stay in the program on the labor force performance of trainees. Moreover, because of the lack of data the estimation of the effect of time on the dependent variables could not be measured. In general, the problem was lack of complete observations on the relevant variables for each of the program participants. Only 112 cases exist of participants with complete information on all of the relevant variables. In the appendix to this study the problem of missing observations in relationship to pooling cross-section and time series data is discussed. We couldn't use pooling techniques because we did not have complete information on the same group of participants over time. And, because of incomplete observations, we couldn't distinguish between different periods. Moreover, as is discussed in the Appendix, the missing observations techniques rely on the incomplete and complete data as well as the underlying theory to fill in for missing components. Our data is anything but reliable and our analytical relationship is basically an empirical and observed relationship, therefore, we cannot give much weight to either method in determining the missing components.

What is important, however, is to note that by defining two variables, one for HOH and one for spouse, each of whom has completed the program, we are estimating the effect of the education of each on the other. Also by using the labor market performance of one as an explanatory variable for the benefit equation of the other we are incorporating the effect of HOH experience on the spouse and vice versa. Although this is not the same thing as incorporating simultaneity in our estimations, it is as far as we can go, given the data conditions we face.

What was done was to use multiple regression to estimate the effect of program participation on the above-mentioned variables while factoring out the effect of all other pertinent variables such as race, state of relocation, family size and so on. Our results are reported in Tables 1 (for HOH), 2 (for spouse) and, 3 (total family income).

With respect to HOH labor market performance we found no significant relationship between the dependent variables and the program variables. In other words we could not reject the null hypothesis that these coefficients were not significantly different from zero. Thus, we found no evidence that program participation by HOH, spouse or both had any significant effect on the labor market performance of HOH as measured by the dependent variables presented in our tables.

With respect to average weeks worked we discover that a spouse who has completed the program has an improved opportunity to work. Compared to the dropout she will work on average about eleven and one-half weeks more. Compared to the spouse of the comparison group member this is even higher. However, the coefficient for the comparison group is not

TABLE 1
The Effect of Program Participation on the Labor Market Performance of Head of the Household*

Program Variables / Dependent Variables	HOH Completor	Spouse Completor	Control	R^2
Average number of weeks employed	2.10 (10.26)	17.78 (11.89)	-26.79 (45.14)	.48
Average monthly income	21.92 (24.08)	-10.93 (28.41)	58.98 (106.49)	.50
Present employment status	-.008 (.14)	-.11 (-.16)	.22 (.61)	.29

* Figures in parentheses are standard errors.

TABLE 2
The Effect of Program Participation on the Labor Market Performance of the Spouse*

Program Variables / Dependent Variables	HOH Completor	Spouse Completor	Control	R^2
Average number of weeks worked	.79 (4.75)	11.49** (5.43)	-30.21 (20.65)	.57
Average monthly income	-18.69 (21.64)	-24.94 (25.39)	206.47** (92.92)	.72
Present employment status	-.004 (.094)	.02 (.11)	.51 (.41)	.56

* Figures in parentheses are standard errors.

** Significant at .05 level of confidence.

TABLE 3
The Effect of Program Participation On Total Family Income*

Program Variables / Dependent Variable	HOH Completor	Spouse Completor	Control	R^2
Total family income	761.67** (376.01)	19.37 (424.26)	1,119.91 (2,345.86)	.27

* Figures in parentheses are standard errors.

** Significant at .05 level of confidence.

significantly different from zero. Therefore, the difference between the spouse of the comparison group member and spouse of completors is the same eleven and one-half weeks. But when we compare average monthly income we find that the spouse of the comparison group member earns on the average 207 dollars more than a spouse who has completed the program. This points to one possible conclusion: That is, the comparison group and the trainees are not from the same population, and, as a result, we have no valid test of program impact. Spouses of the comparison group members earn more income than the spouses of trainees. This can be due to several reasons. But what is important from our point of view is the fact that we have a comparison group whose characteristics are far from making it a reasonable substitute for trainees. From the standpoint of the program it is important to discover those elements in the selection process of trainees and the comparison group which have resulted in their dissimilarity. But this is a matter of administrative check and cannot be pursued here.

Finally, with respect to total family income we find out that families of HOH completors experience high average income, earning 761 dollars more than dropouts. Given the type of people from which trainees and comparison group are chosen, it is hard to believe this outcome. It may have been caused by wrong reporting or it may be that the family income measured was mistakenly interpreted by respondents to be total income over the past six months. Nevertheless, this is an indication, to some extent, that program participation has improved the lot of completors. Note also the fact that although the coefficient of the variable representing the comparison group is not statistically significant, its magnitude is greater than the coefficient for HOH completors. This can be conjectured to have been caused by the diversity among the comparison group members. That is, there are people with extremely high and extremely low income; extreme, that is, compared with the average trainee or comparison group member.

One may wonder why neither husband's nor wife's income was affected by program participation but total family income was affected. As was mentioned above, the quality of data on Mountain-Plains trainees, at least for that available for the pilot study is quite poor. It is highly possible that either of these variables as reported in the questionnaires are misinterpreted by trainees. Also there were questionable practices (now ceased) in transforming variables related to income. We are reasonably sure that this problem stems from the data coding and collection, in that we tried different models and estimated the effects of the program under different conditions and the situation did not change appreciably.

Conclusion

From these results we can conclude that there is some evidence, albeit faint and unreliable, that trainees have benefited from the program. However, the amount of benefit is not clear to us and in fact we cannot ascertain its existence beyond a reasonable doubt.

Our more important conclusion, however, is that we are fundamentally defeated in our efforts because of shortcomings in the data. The reporting of data has been extremely poor and it should be of first order priority to amend the situation. Moreover, the comparison group is not a good substitute for the trainee group. There were apparently problems in the procedure of designating comparison group members. This does not affect the program itself but has serious implications for a valid evaluation of the program. As it stands now, it is a cause of the distortion of results presented in this chapter.

Thus our recommendations in this chapter are exclusively with respect to improving data collection. Without a good system of data collection all future evaluations of the program are doomed to failure. Our specific proposals for data collection are the following:

1. It should be possible to trace the history of each trainee and comparison group member over, say two or three years. Therefore each trainee and comparison group member should be assigned a code number and data booklet which should remain the same throughout the period and all the data pertaining to the trainee whether before, during or after the program, including his or her academic performance, should be recorded in a consistent fashion.

2. The use of such a system allows one to record the socio-demographic characteristics of trainees and the comparison group as well as their pre-program labor market experience.

3. The number of questions asked should be reduced to the minimum necessary level, irrelevant questions or things that it would be "nice to know" should be eliminated. Some may argue that additional information doesn't hurt. The fact of the matter is that it does. Apart from the added time and cost, what is the use of asking hundreds of questions when most of them will go unanswered and those answered are not

reliable? This was the situation with the data we received. Asking too many, and sometimes irrelevant questions, will cause a reduction in the attention of the respondent. He may simply make up some answers to get rid of the interviewer or to avoid difficult questions. Checking for completeness of data is extremely costly and many omissions will go unnoticed. Fewer questions not only will enable more attention to each question but also enable the program administrators to assign a person to assist trainees in answering the questions and at the same time check for the completeness of the questionnaire.

4. The information relating to a couple need not be on the same booklet, but there should be cross reference for identification.

5. It should be determined whether or not the HOH is married and his (her) sex should be recorded.

6. It should be determined whether the respondent has been actively seeking employment during any period. This information will help identify if lack of employment is due to non-participation in the labor force. More-

over it will make it possible to evaluate the effect of the program on labor force participation.

7. Any kind of transformation or tinkering with the data in the process of collection should be emphatically avoided. The investigator should receive raw data to decide for himself how he wants to use it.

8. It is the case that during the life of this program its content and aim changed considerably. These changes apparently ranged from minor modifications to drastic changes resulting in a substantially different program. If these changes are not carefully recorded, the analyst may conduct his analysis on the belief that he is analyzing a program of constant structure over time while in fact he has measured the effect of two or more different programs in his sample. The change in program structure may be unavoidable. The point is that these changes have to be recorded in such a way that each group of trainees are related to the version of the program they participated in. This requires recording of the exact time

of entrance and exist of trainees to and from the program. The data we were given by the Mountain-Plains program was badly deficient in this respect.

9. A related subject is the changes made in the experimental design. If the nature of the data, definition of items, procedure of interview and procedure of recording data is changed, there will be no unified series of observations. Again the analyst who combines all his data for estimation is adding apples and oranges. These changes will result in the samples becoming incomparable over time. Also, it could be that during one period of the program the trainees are given a test at the time of entering and later on the time of the test is moved to one month after entrance. In such a case the meaning of test scores is completely different. The same is true when the test is changed and, therefore, the scores are not comparable. Our recommendation is against changing the experimental design. In cases where such changes are necessary they should be kept to the minimum. But more important, in such cases

the old information should be collected in addition to the new data. For example, if the type of entrance test is changed, it is better to give the trainees two tests (old and new) and record the results of both. Thus, not only will the analyst have a unified and comprehensive series on the result of the old test, but using statistical methods he will be able to extrapolate the results of the new test to get information on the former group of trainees.

10. In the present study we finally ended up with 112 observations. Aside from difficulties arising from this situation, which was discussed in the study, there is the problem of optimal sample size. The sample size should not be left to accident. Also, the composition of sample as to how many comparison group observations are needed compared to the number of trainees is very important. One cannot use two observations on the comparison group in conjunction with a hundred observations on trainees. The results will be badly distorted. Unfortunately these were the problems we encountered in working with the data

collected by the Mountain-Plains program.

11. Finally, the experimental design which was imposed on the original Mountain-Plains program was violated in subtle and not so subtle ways. Thus, the intended control group was only a comparison group. This situation clearly could have been avoided since it would have been possible, given the limited capacity of the facility to have over-subscription to the program. Random assignment, therefore, would not have violated equity considerations.

It is our recommendation that a) short of having observations on all trainees, sampling and the determination of sample size should follow established scientific procedures and, b) the number of control group members, in relation to the number of trainees interviewed, should be determined on the basis of statistical techniques.

CHAPTER THREE

COSTS

In this chapter we are concerned with the cost of operating the Mountain-Plains training center. The analysis is based upon an as yet incomplete set of cost and output data. For a variety of reasons considerable delay in generating these data was experienced. However, rather than delay further we have decided to provide preliminary analysis of these data at this time in order to uncover data and methodological issues.

The first problem in any cost study is to determine whose cost we are considering. It is well known that an economic enterprise may cost differently to different groups of people.[1] Indeed, what may appear to one group of people as cost may be regarded as a benefit by another group. A good example is the stipend paid to the trainees. From the point of view of the Federal Government and by extension of the taxpayers, this is definitely a cost. From the point of view of trainees this is a benefit. From the point of view of the society as a whole it is neither cost nor benefit; it is a transfer payment. Thus, it is important to bear in mind whose cost we are talking about. In the present study we are concerned with the following costs:

---

[1] This subject is discussed in our "Cost Analysis of Manpower Programs: An Analysis of the State of the Art." See pp. 1-3.

Federal Government cost

Individual cost

Social cost

Prototype cost

Federal Government cost represents the total government expenditures on the project. The determinants of this cost and its variations are one of our subjects here. Individual cost is the cost incurred by trainees in order to join the training center. Subtracting transfer payments from the opportunity cost to the trainee of joining the program gives us the individual cost.[2]

---

[2] We use a general definition of opportunity cost which includes foregone earnings as a result of program participation plus direct outlays.

The present chapter, however, does not deal with individual cost. The sum of government cost (net of transfer payments) plus the opportunity cost of trainees is social cost.

Let:

GE = total government expenditures

T = transfer payments

Then that part of government expenditures which is a part of social cost (GC) can be written as

$$GC = GE - T$$

If we denote

OP = earnings foregone by trainees

then individual cost (IC) net of transfer payments will be

$$IC = OP - T$$

We defined social cost (SC) as

$$SC = GC + OP$$

It is easy to see that we can write:

$$SC = (GE-T) + (IC+T) = GE + IC$$

that is, social cost is equal to government expenditures (including transfer payments) plus individual cost (net of transfer payments). The discussion of social cost is left to the time when we have completed the analysis of both government and individual cost.

Prototype cost, simply stated, is the cost of replication of the training center exclusive of research and development costs. Thus, it is different from the total cost of the center in that the Mountain-Plains center engages in research and development activity in addition to training. However, because of the lack of any measure of output for research and development activities at present, this task cannot be performed in an economically meaningful way. We need a measure of research and development activity. The first step in the development of prototype costs is that the Mountain-Plains research and development sections prepare a report to elaborate on the following points:

1. Justification of the research as a part in the whole project.

2. Number of people involved in research, their degrees and experience and the duration of their involvement in man-hours as a function of time, i.e., man-hours/month.

3. Taking only a sample of research and other activities performed, determine a subjective trade off between pairs of these sample projects (i) in terms of input of manpower and equipment (if applicable) and (ii) in terms of its value to the whole project (point 1, above).

The reasons for proposing the formation and measurement of this index will become clear as we proceed to estimate a total cost function for the center.

Types of Cost

To simplify the analysis, the cost data can be broken down into several sub-sets. To begin with, total expenditures consists of a) transfer payments and b) the cost of running the center in terms of net additional resource use (social cost).[3] Thus, on the one hand we will have transfer payments which are related to the number of trainees in a linear way and on the other, that part of the government expenditures which is a part of social cost and is related to the number of trainees but not necessarily in a linear form. Finally, we have the sum of these two functions which form the total government expenditures function.

Before starting our analysis, there is one important point to be made. Our cost figures are monthly data, but as is well known, delays occur between payments for a service rendered and the actual receipt of that service. Therefore, due to time lags, monthly figures may show drastic instability when analyzed. To overcome this problem we analyzed the data by means of a moving average. The moving average was obtained as

---

[3] For a discussion of the subject of cost, expenditures and transfer payments, see our "Cost Analysis of Manpower Programs: An Analysis of the State of the Art."

$$\tilde{X}_t = \sum_{i=0}^{2} X_{t-i}$$

where X_t is the monthly cost figure and $\tilde{X}_t$ the moving average for output. Note that we chose a three-month period on the basis of an arbitrary judgment that it usually wouldn't take more than three months to pay for an item or a service.

Transfer payments. A transfer payment is the payment to another party whenever a reciprocal exchange of goods or services is not expected. From the definition it is clear why we would like to study transfer payments separately and to exclude them from costs. Transfer payments are not a necessary part of costs needed to run the center. Thus, they are not a part of the social cost of a replica of the center if the analysis is to determine the economic desirability of having similar centers. Moreover, in almost all cases transfer payments are set arbitrarily and they are not related to output in any specific way. Thus, their inclusion in the social cost in a cost-benefit analysis aimed at determining the rationality of continuing the project, the optimal level of output or the desirability of the expansion of the project, is meaningless and will distort the conclusions concerning project efficiency. However, from the point of view of the government and the taxpayer (assuming taxpayers are distinct from the recipients of the transfer), it is a part of the program cost. Nevertheless, both government and taxpayer do not view a transfer payment in the same light as they do other expenditures. Therefore, it is advisable to treat these two categories of expenditures separately. The transfer payments can be to another government agency or to individuals. In the context of the present study, there is no intra-government transfer payment, because the base is rented for one dollar per year. However, if there was any rent paid, we would have considered it a transfer payment. This may seem strange to treat the money paid by the NIE to the Department of Defense as a transfer payment. One may argue that this would be the rent paid for the use of buildings and facilities belonging to the Department of Defense. As we have argued elsewhere,[4] this could be true if the Department had foregone an opportunity to use the base. From the inspection of the buildings and its location and considering the fact that many buildings remain vacant, one has no recourse but to conclude that the opportunity cost of the use of the base by Mountain-Plains is practically zero.

Note, however, that if it is decided to replicate this center in another location where such facilities are not available, the prevailing rent of a suitable place should be considered as a part of the cost and be added to the measures of social cost we compute here.

It has to be noted that Mountain-Plains does make payments to the Department of Defense. But these payments are in return for fire and security protection, snow removal, recreation and utilities and maintenance--additional resources used up by society due to the presence of the program. To gain an understanding of the magnitude of these payments, Table 1 shows the amount of payment for the monthly periods of September 1973 through June 1974.

---

[4] "Cost Analysis. . . ," op. cit., pp. 3-9

These payments are of special importance. To begin with, the relationship between these figures and the cost to society and the government is a complicated one. While the production cost of the services performed by the Department of Defense may be very much lower than these figures, the price of obtaining them in the local market may be much higher. Moreover, a part of the cost for office buildings goes for research activity that cannot be included in the prototype cost. This is a rather involved problem, and we will come back to it later in this chapter.

Transfer payments to the trainees. Like transfer payments to a government agency, the stipend and any

TABLE 1

N.I.E. Payments to the Department of Defense, in Dollars

Date	For Office Building	For Student Housing	Total
September 1973	15103	22702	37805
October 1973	21090	21985	43075
November 1973	22662	24554	47217
December 1973	44121	23141	67263
January 1974	24248	30820	55068
February 1974	22538	32006	54544
March 1974	29334	21304	50638
April 1974	23369	32858	56227
May 1974	26810	40303	67113
June 1974	37825	31980	69804

other transfer payments (including payments in kind) to trainees are not a part of the social cost of running a center and therefore have to be excluded from the cost data and studied separately. In the Mountain-Plains project we have four cost items which can be subsumed under transfer payments to trainees. These four categories are: student stipend, day care assistance, insurance (students) and medical costs (students).

Table 2 presents total transfer payments to the students during each month (Column 2). Column 3 shows total "family months" in the center, that is, the number of families enrolled in the center during the whole month plus those staying for a fraction of the month multiplied by that fraction. (These data were taken from the list of entering and exiting students.) In Column 4 we have the actual number of families who stayed in the center at some time during each month. These data are included here for the sake of comparison with the data in Column 3. Finally, Column 5 gives the average monthly transfer payments per family.

TABLE 2

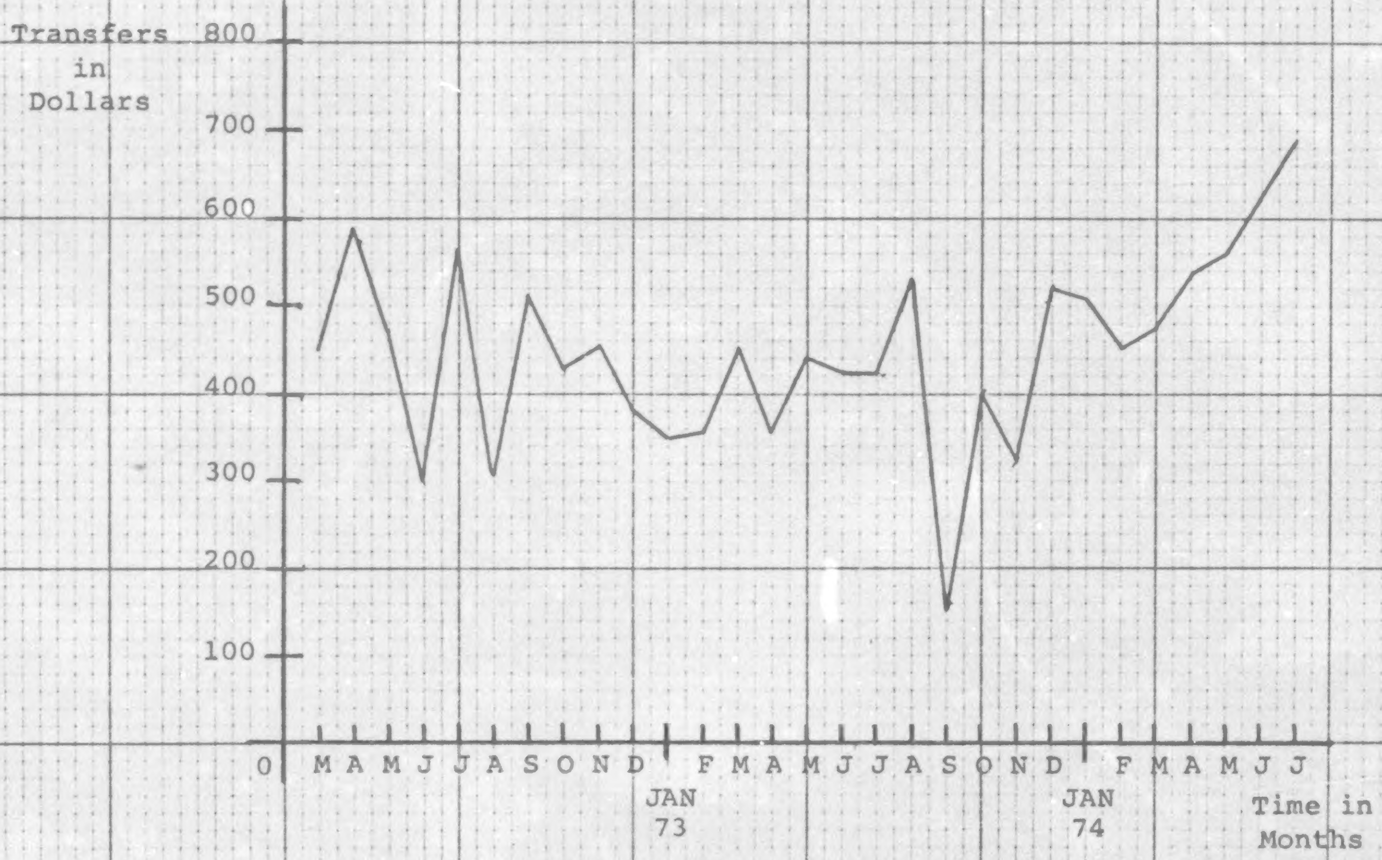
Transfer Payments to Trainees, in Dollars

Date	Total Transfer Payments	Family Month	No. of Families in Center	Average Monthly Transfer Payments	Total Stipend	Average Stipend per Family Month
March 1972	4902	11		446	4902	446
April 1972	5287	9		587	4007	445
May 1972	10821	23		471	10821	471
June 1972	12784	43		297	11026	256
July 1972	38938	67		581	20751	310
August 1972	28655	93		308	28620	308
September 1972	52653	103		511	42528	413
October 1972	45118	105		430	41139	392
November 1972	53664	120		447	48942	408
December 1972	51433	136		378	45143	332
January 1973	51593	146		353	45080	309
February 1973	58942	161		366	51381	319
March 1973	78893	176		448	68767	391
April 1973	67702	184		368	58148	316
May 1973	80336	178		451	70689	397
June 1973	76862	173		444	51991	301
July 1973	68969	163		423	57879	355
August 1973	87401	165		530	75648	459
September 1973	25618	169	174	152	17326	103
October 1973	68543	173	178	396	57602	333
November 1973	56892	173	191	329	54865	317
December 1973	96909	174	206	557	76138	438

TABLE 2 - TRANSFER PAYMENTS TO TRAINEES, IN DOLLARS -- (Cont'd)

Date	Total Transfer Payments	Family Month	No. of Families in Center	Average Monthly Transfer Payments	Total Stipend	Average Stipend per Family Month
January 1974	91134	179	209	509	77717	434
February 1974	82098	181	210	454	64223	355
March 1974	85175	179	204	476	62837	351
April 1974	95295	178	202	535	79091	444
May 1974	98221	175	204	561	78885	451
June 1974	71216	163	188	437	40361	248
July 1974	115590	168	191	688	68676	409

FIGURE 1 - AVERAGE TRANSFER PAYMENTS TO TRAINEES



SOURCE: TABLE 2

It has to be noted that the difference between our figures and those of the Mountain-Plains accounting office arises from the fact that we consider as our output measure the number of "family months" in the center, that is, the total number of families who have stayed in the center for the whole month plus the number of families who stayed a part of the month multiplied by the fraction of the month they stayed. As is clear, using the total number of families in the center as an output measure is less desirable.

As can be seen, the average transfer payments fluctuate dramatically from a low of about $152 per family month (September 1973) to a high of $688 (July 1974). This wide range of average transfer payments cannot be explained by any reasonable assumption based on the production of skills at the Mountain-Plains project. The explanation for this is that fluctuations in the last three items--namely, day care assistance, insurance and medical costs--have caused this pattern of change. Column 6 of Table 2 gives the total stipend and Column 7, the average stipend per family. As can be seen, the fluctuation is reduced considerably to the range of from 103 (September 1973) to 471 (May 1972).

Costs. Having excluded transfer payments, we now have to deal with all other costs. These costs fall into three categories: variable cost that can directly be related to certain trainees, overall variable cost and capital cost.

Recruitment/Follow-up and Relocation Costs. Among variable costs, recruitment/follow-up costs can be directly related to completers and relocation cost can be attributed to new plus leaving families. Tables 3 and 4 give some information on these costs.

Recruitment/follow-up costs are exactly as stated, the cost of recruiting students and the comparison group members for the program, collecting the initial sociodemographic data on them, and following them up after they leave the program. Column 2 in Table 3 gives total recruitment/follow-up cost; Column 3 has an involved computation where each completer is considered to take part in any interview the month after his completion (three weeks follow-up), three months after completion (three months follow-up), and six months after completion and every six months thereafter. Column 4 gives the average recruitment/follow-up costs, using project completers as the denominator.

TABLE 3

Total and Average Recruitment/Follow-Up Cost (Monthly Data), in Dollars

Date	Total Cost	Total Interviewees	Average Cost
October 1972	175	2	88
November 1972	113	5	28
December 1972	858	5	172
January 1973	1346	5	269
February 1973	448	9	50
March 1973	1738	7	248
April 1973	2381	12	198
May 1973	3277	20	164
June 1973	3930	25	157
July 1973	2440	26	94
August 1973	1633	29	56
September 1973	448	33	14
October 1973	3605	36	100
November 1973	1372	41	33
December 1973	1396	36	39
January 1974	123	38	3
February 1974	1687	33	51
March 1974	2185	37	59
April 1974	3191	43	74
May 1974	2551	62	41
June 1974	4430	67	66

TABLE 4

Total and Average Recruitment/Follow-Up Cost (Moving Average), in Dollars

Date	Total Cost	Total Interviewees	Average Cost
December 1972	382	4.00	96
January 1973	772	5.00	154
February 1973	884	6.33	140
March 1973	1177	7.00	168
April 1973	1522	9.33	163
May 1973	2465	13.00	190
June 1973	3196	19.00	168
July 1973	3216	23.67	136
August 1973	2668	26.67	100
September 1973	1507	29.33	51
October 1973	1895	32.67	58
November 1973	1808	36.67	49
December 1973	2124	37.67	56
January 1974	964	38.33	25
February 1974	1069	35.67	30
March 1974	1332	36.00	37
April 1974	2354	37.67	63
May 1974	2642	47.33	56
June 1974	3391	57.33	59

As can be seen, the average recruitment/follow-up cost fluctuates extremely from $3 (January 1974) to $269 (January 1973) for each observation. We therefore computed the same average cost using a moving average as described at the beginning of this chapter. Figures in Table 4 show more stability but still the range is from $25 (January 1974) to $189 (May 1973).

We were not able to discern the exact reasons for this fluctuation. It may be that not all persons have been uniformly accounted for over time. It could also reflect changes in the geographic source of trainees and members of the comparison group over time.

With respect to relocation cost the situation is more straightforward. When we take the total number of new families plus leaving families as the denominator, the average monthly cost fluctuates between $93 and $156 (September 1973 and August 1972) to as high as $1,251 and $2,719 (April and March 1972). Put in simple terms, it may cost between $100 and $2,700 to relocate the average family.

Using moving averages, with the exception of the first four months, we get more stable figures, this time ranging from $214 (April 1973) to $320 (November 1973) which is a plausible range. However, the expenses in the first four months remain inexplicable in conceptual terms.

Variable, Capital, and Total Cost. It is important to make a distinction between variable and capital cost. The reason is that the services rendered by physical capital are not exhausted in the period in which the capital item is originally acquired. Rather, its services are spread over the economic life span of the capital good. This period may range from one hundred years (a dam) to thirty years (a building) to ten years (machinery) to one year (tools), to show some arbitrary examples. There are several ways of prorating the cost of capital over its economic life span.[6]

---

[6] See "Cost Analysis . . . ," op. cit., pp. 3-9

In the present study, however, the capital items are restricted to trainee equipment, staff equipment and remodeling of buildings. These items are not important compared to variable costs, and the results will not be very different whether or not they are prorated. In the following we make the distinction between variable cost and total cost (variable plus capital costs), but no proration will be attempted.

Table 7 summarizes the information on variable and total cost. Columns 2 and 3, respectively, are variable and total cost of the center, Column 4 gives the data on total "family months" in the center, and Columns 5 and 6 are, respectively, average variable and average total costs. A glance at the table shows that there is no pattern in these average costs. This is not surprising because these figures are almost conceptually meaningless. The reason for this is that we don't know how much of this cost should be attributed to the research activities of the center. This is an important problem that cannot be solved by a simple proration. Note that if an arbitrary rule of thumb like prorating 40 percent

TABLE 5

Trainee Relocation Costs (Monthly Data), in Dollars

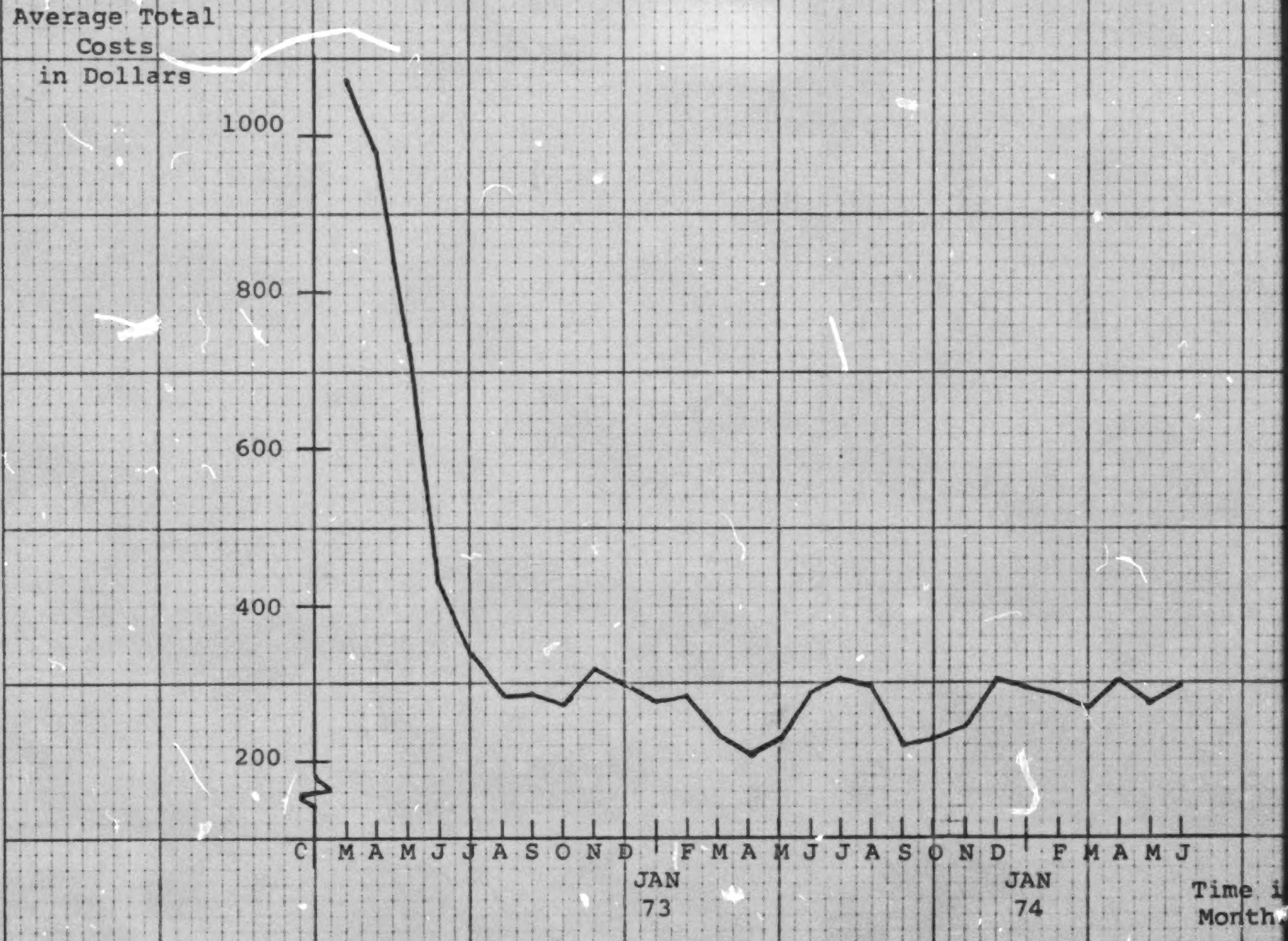
Date	Total Relocation Cost	Average Relocation Cost
January 1972	3134	--
February 1972	4487	374
March 1972	8158	2719
April 1972	10013	1252
May 1972	8473	339
June 1972	8093	279
July 1972	10093	404
August 1972	5159	156
September 1972	2464	493
October 1972	10326	382
November 1972	8669	248
December 1972	84961	274
January 1973	7094	308
February 1973	8269	267
March 1973	11964	196
April 1973	6711	197
May 1973	13375	382
June 1973	7387	274
July 1973	10926	280
August 1973	13568	308
September 1973	4454	93
October 1973	13514	282
November 1973	15241	363
December 1973	7553	252
January 1974	7863	246
February 1974	8574	390
March 1974	9927	242
April 1974	15861	311
May 1974	17688	260
June 1974	22073	324

TABLE 6

Trainee Relocation Costs (Moving Averages), in Dollars

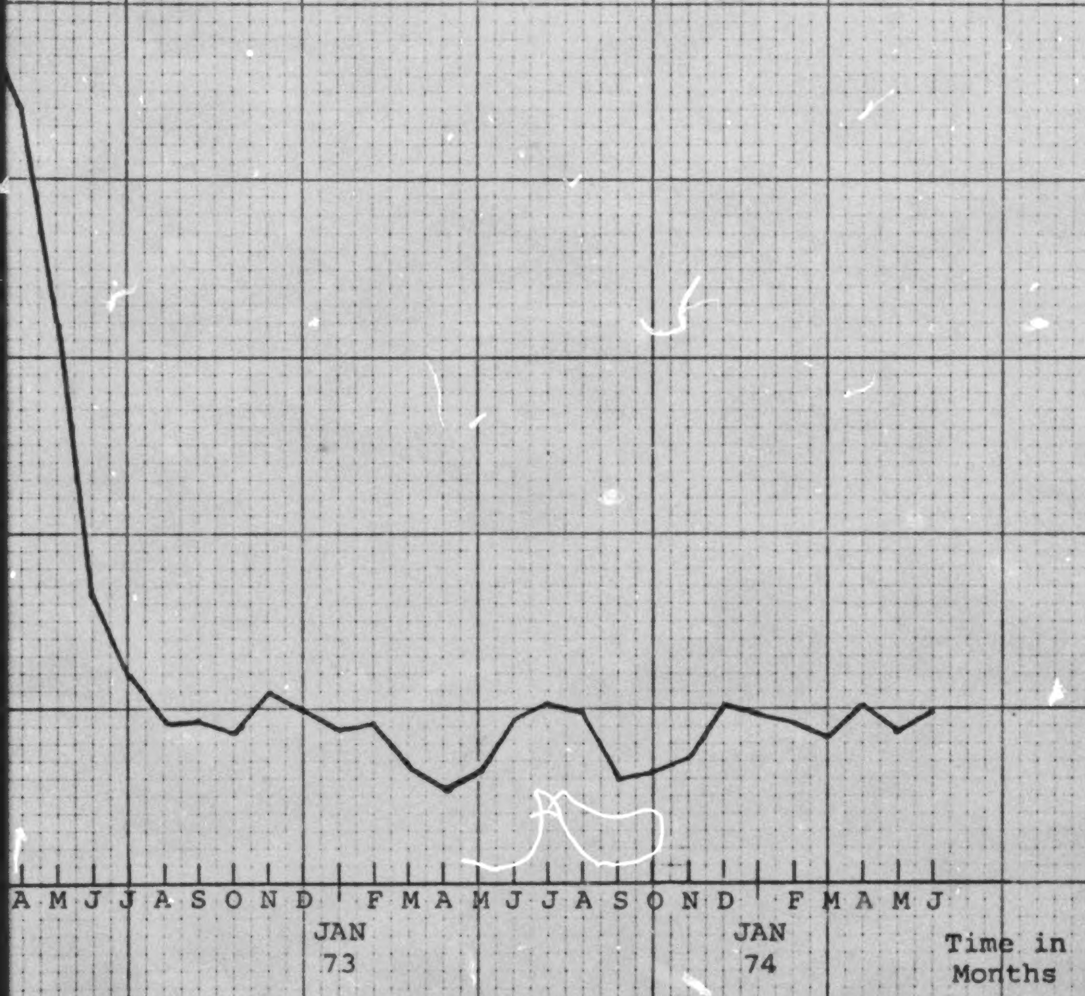
Date	Total Relocation Cost	Sum of New and Leaving Families	Average Relocation Cost
March 1972	5260	5.00	1052
April 1972	7553	7.67	985
May 1972	8881	12.00	740
June 1972	8860	20.67	429
July 1972	8886	26.33	337
August 1972	7782	29.00	268
September 1972	5905	21.00	281
October 1972	5983	21.67	276
November 1972	7153	22.33	320
December 1972	9164	31.00	296
January 1973	8086	29.67	273
February 1973	7953	28.33	281
March 1973	9109	38.33	238
April 1973	8981	42.00	214
May 1973	10683	43.33	247
June 1973	9158	32.00	286
July 1973	10563	33.67	314
August 1973	10627	36.67	290
September 1973	9649	43.67	221
October 1973	10512	46.67	225
November 1973	11070	46.00	241
December 1973	12103	40.00	303
January 1974	10221	34.67	295
February 1974	7998	28.00	286
March 1974	8790	31.67	278
April 1974	11454	38.00	301
May 1974	14492	53.33	272
June 1974	18541	62.33	297

FIGURE 2 - AVERAGE RELOCATION COST (Moving Average Data)



SOURCE: TABLE 6

IGURE 2 - AVERAGE RELOCATION COST (Moving Average Data)



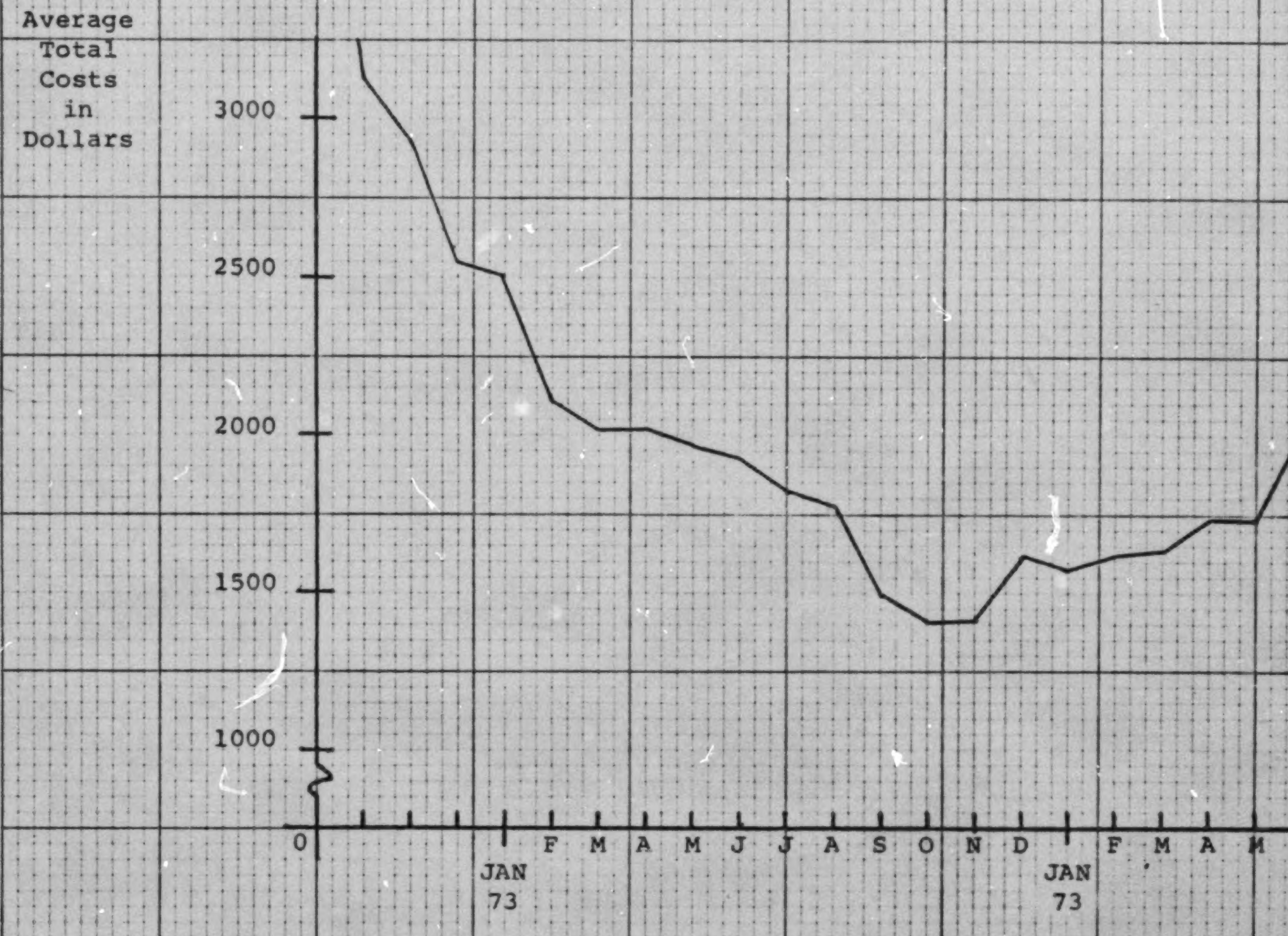
URCE: TABLE 6

TABLE 7

Variable and Total Costs (Monthly Data), in Dollars

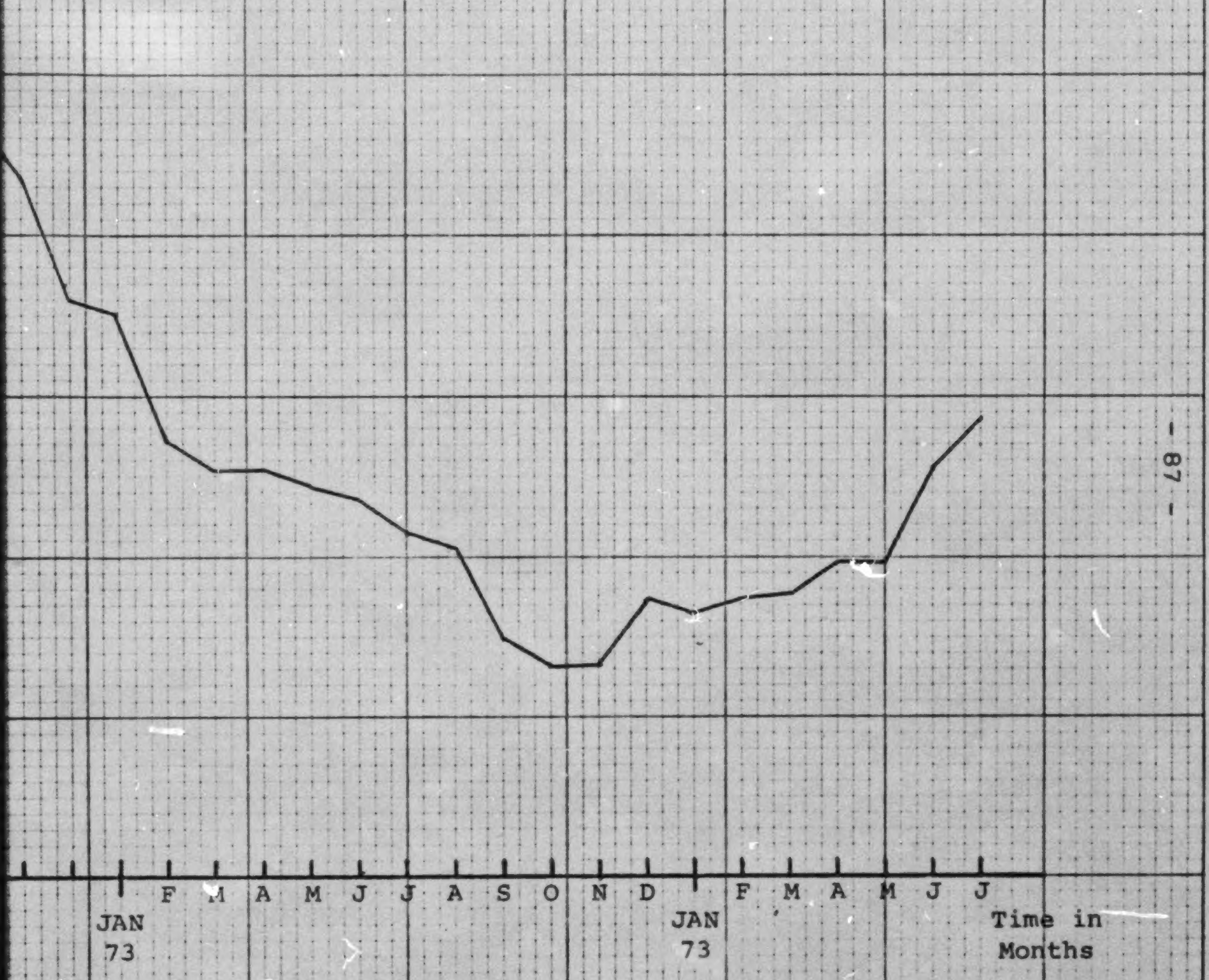
Date	Variable Costs	Total Costs	Family Months	Average Variable Cost	Average Total Cost
January 1972	171561	206258	0	---	---
February 1972	253298	296752	1	253298	296752
March 1972	206285	236285	11	18753	21481
April 1972	337412	355944	9	37490	39549
May 1972	387676	473680	23	16856	20595
June 1972	266021	298457	43	6187	6941
July 1972	294819	330856	67	4400	4938
August 1972	306522	342758	93	3296	3686
September 1972	302602	336597	103	2938	3218
October 1972	193908	263024	105	1847	2505
November 1972	307516	359406	120	2563	2995
December 1972	266485	298972	136	1959	2198
January 1973	303311	348468	146	2078	2387
February 1973	241502	288475	161	1500	1792
March 1973	335173	336546	176	1904	1912
April 1973	420302	425676	184	2284	2314
May 1973	281819	295230	178	1583	1659
June 1973	300696	311922	173	1738	1803
July 1973	318564	330099	163	1954	2025
August 1973	249033	253669	165	1509	1537
September 1973	157674	158213	169	933	936
October 1973	294655	300646	173	1703	1738
November 1973	260327	268036	173	1505	1549
December 1973	271012	271051	174	1558	1558
January 1974	285681	294972	179	1596	1648
February 1974	289102	291144	181	1597	1609
March 1974	291332	291935	179	1628	1631
April 1974	339856	344078	178	1909	1933
May 1974	253846	262466	175	1451	1500
June 1974	428204	440426	163	2627	2702
July 1974	339766	397773	168	2022	2368

FIGURE 3 - AVERAGE TOTAL COSTS (Moving Average Data)



SOURCE: TABLE 8

IGURE 3 - AVERAGE TOTAL COSTS (Moving Average Data)



CE: TABLE 8

of costs to research activities is adopted, then the result will not be very different from the figures in the table. It is enough to multiply them by .6. The wide fluctuation still remains and is unexplained though our guess is that it reflects the fact that the experimental model (its educational production function) was changed a number of times over the course of the experiment. Thus, the issue is deeper than one of arbitrary proration, and the next section is devoted to a discussion of this problem and our recommendation as to its possible solution.

Although by using the moving average method the fluctuations decrease (Table 8), the above problem remains an obstacle to the meaningful interpretation of the data.

Having finished an examination of descriptive data, we turn now to the analysis of the cost functions.

Cost Function

In this section we analyze the cost function of the Mountain-Plains project and try to point out why a meaningful total cost function cannot be estimated.

TABLE 8

Total and Average Total Costs (Moving Averages), in Dollars

Date	Total Cost	Family Months	Average Cost
March 1972	246432	4.00	61608
April 1972	296327	7.00	42332
May 1972	355303	14.33	24794
June 1972	376027	25.00	15041
July 1972	367664	44.33	8294
August 1972	324024	67.67	4788
September 1972	336737	87.67	3841
October 1972	314126	100.33	3131
November 1972	319676	109.33	2924
December 1972	307134	120.33	2552
January 1973	335615	134.00	2505
February 1973	311972	147.67	2113
March 1973	324496	161.00	2015
April 1973	350232	173.67	2017
May 1973	352484	179.33	1966
June 1973	344276	178.33	1931
July 1973	312417	171.33	1823
August 1973	298563	167.00	1788
September 1973	247327	165.67	1493
October 1973	237509	169.00	1405
November 1973	242298	171.67	1411
December 1973	279911	173.33	1615
January 1974	278020	175.33	1586
February 1974	285722	178.00	1605
March 1974	292684	179.67	1629
April 1974	309052	179.33	1723
May 1974	299493	177.33	1689
June 1974	348990	172.00	2029
July 1974	366888	168.67	2175

Then we will try to suggest a way out of this problem.

Economic theory prescribes that cost is a function of output. But the main part of the theory is concerned with the cost function when the firm (or any production unit) is in equilibrium.[7] But what happens when the firm is not in equilibrium? Indeed, what is the nature of the problem when the firm has no idea of equilibrium and is groping towards it? If we had many similar centers which had been in operation for a period as long as, say, five years, we could have aggregated the data over time and estimated a cost function from this set of cross-section observations.[8] But the Mountain-Plains project is a unique center. We have observations on its operation over a period of time during which it hasn't been in equilibrium; rather, as a normal consequence of its development it has been groping towards its normal mode of operation for some time.

---

[7] For a discussion of cost functions, see "Cost Analysis. . . ," op. cit., 13-15.

[8] This was the practice adopted in our An Econometric Analysis of the Costs of Selected Manpower Programs.

How do we estimate the total cost function in such a case?

If a firm was in equilibrium for some time, then output would remain constant for this period, and therefore one could safely relate the current cost to current output. But if a firm is moving from one level of activity to another in search of equilibrium, then there will be some adjustment cost, and the more abrupt the adjustment, the higher the associated adjustment cost. In other words, the cost to the firm not only depends on the firm's current output, but on its past position and in fact on its path of output adjustment during the recent past.

Thus, we can assume that the cost of each month depends on a weighted average of the output of the current month and several past months. The meaning of this relationship can become even clearer by noting that the cost of each month may be based on the planned or expected output plus some adjustment for actual output. If expected output is a weighted moving average of the past months, then we can write the cost function as follows:

$$(1)\quad C_t = a + b \sum_{i=0}^{\infty} \lambda_i X_{t-i} + u_t$$

where

C_t = cost at time t

X_t = output at time t

a, b and λ_i = constants and

u_t = an error term.

If we assume that λ_i's follow a geometrically declining pattern, then we can give still another interpretation to this cost function.

Let

$$\lambda_i = \lambda^i \qquad \text{for } i = 0 \ldots \infty$$

Then we can write

$$(2)\quad C_t = a + b \sum_{i=0}^{\infty} \lambda^i X_{t-i} + u_t$$

and

$$(3)\quad \lambda C_{t-1} = a \lambda + b \sum_{i=1}^{\infty} \lambda^i X_{t-i} + \lambda u_{t-1}$$

subtracting (3) from (2) we have

$$(4)\quad C_t = (1- \lambda)a + \lambda C_{t-1} + b X_t + (u_t - \lambda u_{t-1})$$

The interpretation of this function is simple. The cost of each month is a percentage (possibly more than 100 percent) of the past month cost adjusted for the current month output. In other words the firm takes the past month cost, adds (subtracts) some percentage to (from) it and allocates the cost for the current month. This cost is adjusted in the course of the month for the current rate of the output. This form of the total cost function is called the autoregressive form of a distributed lag function.

The meaning of adjustment cost can be explained easily by an illustration: Consider a training center where, say, two hundred students are currently getting instruction. With no other change, if we increase the number of students by 5, 10, 20 and 25, there may be little or no additional cost, therefore, leaving the average cost intact or slightly affected. This is because a few additional students can be accommodated easily in the existing classes with present instructors and can use essentially the same instructional equipment. Note that here we do not assume that the center is working exactly at its capacity. In short, marginal cost is not much higher than average cost.[10]

In the case of the training center, a doubling of the number of students will have drastic effects on costs. And, the larger the number of additional students, the larger this effect. If the center is not going to lower the quality of its service, this increasing additional increment to total cost arises from the fact that competent instructors may not be available and the center may have to attract them with higher cost from other locations. Since the center is not geared to processing an infinitely larger number of students, an abrupt increase may result in over-straining equipment, building and instructors, with the accompanying additional costs.

---

[10] It is interesting to note that in a private firm the same thing holds. If the firm under, say, the conditions of monopolistic competition is working below capacity, marginal cost may be constant. However, as capacity is approached, marginal cost starts rising and the closer it gets to its capacity, the higher the marginal cost.

So it is clear that there is a cost of adjustment itself.[11]

One can either estimate the moving average form, *i.e.*, equation (1), or the autoregressive form, equation (4). It can be shown that ordinary least squares applied to equation (4) will result in biased and inconsistent estimates of the coefficients.[12] We estimated several versions of equation (1) both for variable and total cost. Then we chose the best equation among them.[13] This equation was the estimated total cost function. Table 9 gives the details.

---

[11] A similar case of adjustment cost, the one related to the time between decision to produce and production, is discussed by Alchian. See A. A. Alchian, "Costs and Outputs" in M. Abramovitz, *et al.*, *The Allocation of Economic Resources: Essays in Honor of Bernard F. Haley*. Stanford, California: Stanford University Press, 1959.

[12] There are alternative methods of estimation which are unbiased and consistent. In view of the quality of data and the cost of such estimation in terms of programming, it was decided not to attempt them at present.

[13] For a discussion of our strategy for choosing among the estimated equations, see our *An Econometric Analysis of the Costs of Selected Manpower Programs*, Appendix A.

TABLE 9

Estimates of the Coefficients of the Cost Function

Variable	Estimated Coefficient (Standard Error)	t - statistic
Intercept	371648	
X_t	443.75 (2332.43)	.19025
X_{t-1}	-3669.22 (3775.20)	- .97193
X_{t-2}	5657.45 (3888.59)	1.45488
X_{t-3}	-3192.90 (3574.36)	- .89328
X_{t-4}	368.85 (1803.19)	.20455
R^2	.4328	
No. of Observations	27	

There are problems in applying the ordinary least squares technique to the estimation of this model. The estimates will be biased and inconsistent. To avoid this and get estimates of the parameters of the model with desired properties of unbiasedness and consistency we use the search technique. With use of this technique the estimator is maximum likelihood and other estimates are unbiased and have the least variance in the class of linear estimators. However, there are difficulties with this technique which arises from the specific form of the model that we have adopted here.[14]

Using this technique in estimating the cost function the result was as follows:

$$y_t = 125490 + .68\, y_{t-1} - 160.85\, X_t$$

$$\qquad\qquad (10.55) \qquad\quad (1.98)$$

$\bar{R}^2 = .99$, number of observations = 29

where the figures in parentheses are t-ratios.

---

[14] The problem of inconsistency of the ordinary least squares as well as detailed discussion of search technique and its statistical properties together with a program for search estimator are presented in the econometric appendix to this study.

From the equation estimated the following conclusions can be drawn.

(1) $\bar{R}^2$ is high and therefore we are on a firm ground in that we have explained a considerable portion of the variation in monthly cost.

(2) There is a simple past-month-budget-review scheme in the expenditures.

(3) The coefficient of output has turned out to be negative. But this variable is not significant in explaining the variation in total cost. In other words the coefficient is not significantly different from zero, which means that output has no effect on cost. This in turn strengthens the conclusion stated under point (2) above. The conclusion is obvious. The program is set up with certain capacity to process trainees and what is done is simply to revise the average last three months budget and spend it for the next month. Of course this is true with due consideration to the lags between committing the funds and expenditures. In other words the proposition is correct in a moving average sense. Our attempt, therefore, to find a relationship between cost and output failed.

Now this conclusion is in conflict with the conclusion drawn from the moving average form of the function. The reader may argue, also, that the two were mirror images of each other and one could have been deduced from the other. These objections, however, can be answered satisfactorily. First, it is true that the two functions can be deduced from each other but at the same time each of them can be set up without reference to the other, and in such a situation the implication for the error term will be different. Indeed our estimates point this out. The first estimate implies a negative λ and the second a positive λ. Which of the two should we choose? We believe the second one, and for the following reasons:

1. The estimation of the first equation is marred by the statistical problems mentioned above.

2. The first equation explains only 43 percent of the variation in the cost while the second equation explains 99 percent.

3. Not all the variables in the first equation are statistically significant.

4. And finally we have to consider the assumptions underlying each model. For the first equation we assume that the firm or the unit under investigation deliberately adjusts its output (in this case trainee accommodation) and costs in order to move towards the optimum (lowest average cost) point of operation. The second equation is based on a single budget-revision assumption. That is, every month the unit takes it past period budget and makes a simple revision of it. It seems that the underlying assumption for the second equation comes closer to our a priori information with respect to the operation of such projects.

Thus it seems that the natural choice is the second equation, the autoregressive form. By now it should be clear that we can no longer talk of a cost function in the sense of Chapter One above. That is, we can no longer rely on the hypothesis that there is cost minimizing behavior (subject to any given level of output) on the part of the administration of the program, neither in an absolute sense nor subject to an information constraint which results in the search for an optimum level of output and a partial adjustment to the discrepancy between the actual and the desired (optimum) level of output.

With respect to our work this implies that we cannot estimate the program cost function but only the government expenditure function.

Total Government Cost

Now that we have completed the analysis of each component of the total government expenditure function, we can put them together to form the total government expenditure function. Note that this function is subject to all the qualifications of its parts and it reflects the true total government expenditure function only to the extent that its parts are representative of each part of government cost. The function will be:

$$y_t = 125490 + .7\, y_{t-1} + 660\, X_t$$

In other words aside from a constant term the expenditure function of the program for the three consecutive months including the present month is equal to 70 percent of the same variable (for the past three months) plus $660 for each trainee-month in the center during the present month. This last part is a direct cost attributable to trainees and is related to their numbers in a direct linear way.

Conclusion

The basic conclusion of this chapter is that it is next to impossible to perform a classical cost-benefit analysis of a program when it is still in a state of flux and development. As the program changes, the structural relationship between program inputs and the desired outputs changes, often in unclear and unspecified ways. Changes in the Mountain-Plains project appear to have been continuous over most of the period for which this pilot study was conducted and the exact nature of the changes were not always well documented. Even if they had been well documented, too few observations exist for any one period of short-run stable operation to allow a statistical evaluation of project costs and benefits.

However, apart from this fundamental problem, which is reflected most clearly in the cost data, we should report that the quality of the financial cost accounting data was adequate to perform the required cost analysis had the project been in a stable state of development over the period for which most of the data were available.

APPENDIX A

Econometric Issues

This Appendix is not intended to be a comprehensive treatment of econometric issues involved in cost benefit analysis. Rather it is included in this study to present a detailed account of econometric problems that we encountered in this study or that we could have encountered had we had a more comprehensive and more reliable set of data. Also it discusses the consequences of such problems and solutions we found for them. As such it is hoped to serve as a guide for the future studies of the program under study.

Specification Analysis

In any cost benefit analysis the specification of the model to be estimated is a problem to face squarely. What are the variables to be included (including different powers of a variable), and what is the proper functional form of the equation?

The set of possible variables and their combinations is unlimited. The constraints on these sets are economic theory, availability of data and prior information. These should reduce the number of prospective variables to a reasonable size. Now, the question is which of these should be retained and which should be omitted. Needless to say, if the estimates of the coefficients of a set of variables are not statistically significant, there is no point in retaining them. How do we determine this? Consider the following model:

$$y = X\beta_1 + W\beta_2 + u = [X \; W]\begin{bmatrix}\beta_1 \\ \beta_2\end{bmatrix} + u \quad X^* \beta + u$$

y is T x 1, X is T x k and W, T x j observation matrices. We want to test the the hypothesis $\beta_2 = 0$. Form the following model:

$$y = X\beta_1 + u$$

where

$$X = X^* \begin{bmatrix} I \\ 0 \end{bmatrix}$$

Form the ratio:

$$F = \frac{(e'e - e^{*\prime}e^*) \,/\, (trM - trM^*)}{e^{*\prime}e^* \,/\, trM^*}$$

where $M^* = I - X^*(X^{*\prime}X^*)^{-1}X^{*\prime}$ $\quad M = I - X(X'X)^{-1}X'$

and $\quad e^* = M^*U \quad e = Mu$

This ratio is distributed as F with (trM - trM*) and trM* degrees of freedom.[1] Thus the null hypothesis that $\beta_2 = 0$ is rejected if the computed F is greater than the critical value of F for the degrees of freedom.

Using this test we can reduce the number of variables to the point where all of them are statistically significant. Note that the theory, prior information and the t-statistic play an important role in choosing the subsets to be tested first.

Note also that after specifying our model in this way, if we use the same set of data to estimate and test the coefficients, the sampling distribution of estimators will be affected. This will not be a problem, however, if the estimation and testing are not based on the same set of data used for specification analysis. Thus, Theil recommends "to divide the available observations into three parts, the first of which is used for the choice of the specification, the second for the estimation, and the third for conditional prediction based on the estimated equation in order to verify whether the method actually works.[2] Now in most cases the economists are faced with a limited number of observations and dividing them into three parts will result in doing "three poor jobs." In order to overcome this problem we have pooled the cross-section and time-series data. This procedure has important implications for estimation that are discussed in the following section.

---

[1] For a proof of this theorem, see Fisher, Franklin M., "Tests of Equality Between Sets of Coefficients in Two Linear Regressions: An Expository Note," *Econometrica*, vol. 38, no. 2 (March 1970), pp. 361-366.

Pooling Time-Series and Cross-Section Data

The reason for pooling cross-section and time-series data is to broaden our data base.[3] In the context of the present study if we take each individual trainee in turn we are going to have at most six observations recorded before entering the program and in certain intervals during the period immediately after completing the program. This is a very small sample. Moreover we are not interested in the performance of one individual; we want to appraise on the average the effect of the program on trainees. On the other hand if we take the observations on all of the trainees in a given date, say six months after completion of the program, we cannot trace the time path of the effect of the program on their labor market performance. The way out of this problem is to pool our cross-section and time-series data.

Now, consider our model:

$$y_{it} = \sum_{j=1}^{P} \beta_j X_{jit} + u_{it}$$

---

[2] Theil, Henry, Principles of Econometrics, John Wiley and Sons, Inc., New York, 1971, p. 603.

[3] (See footnote 4 on next page.)

In order for the OLS estimator to have its desired properties when applied to this model, not only have we to assume that the β_j's are the same for all observations at all times, but we also have to assume that the net effect of all other influences left out of the model, as reflected in the error term, has the same mean and variance for all observations over time. This is an unrealistic assumption. A better model is the following:

$$y_{it} = \sum_{j=1}^{P} \beta_j X_{jit} + u_i + v_t + w_{it}$$

That is, we assume the error term to be composed of three components-- one changing over individuals but remaining constant over time for each individual, a second component changing over time but being fixed for all individuals in any one period, and, finally, a remainder.

Note that this is a special case of random coefficient model where only the constant term has stochastic parts. Letting X_1 (the first column of observation matrix X) have all its elements equal to 1, we can write:

$$y_{it} = (\beta_1 + u_i + v_t) + \sum_{j=2}^{P} \beta_j X_{jit} + w_{it}$$

This leads to the natural thought that assuming all β_j's to be stochastic is a superior specification. This latter model was not adopted in this study. Aside from computational difficulties, random coefficient models have not been studied as widely as the simpler model for the pooling of cross-section and time-series data have been.[5]

---

[4] Another use of pooling data is to estimate the coefficients of the variables which change over the cross-section sample while the variables which change over time are constant and then to incorporate these estimates into the time-series data and estimate the coefficient of variables which change over time. (See, for example, Klein, Lawrence R., A Textbook of Econometrics, 2nd ed., Prentice Hall, Inc., Englewood Cliffs, N. J., pp. 374-382.

[5] For the discussion of random coefficient models, see Swamy, P. A. V. B., Statistical Inference in Random Coefficient Models, Springer-Verlag, New York, 1971; Swamy, P. A. V. B., "Linear Models with Random Coefficients," in Zarembke, P., (ed.), Frontiers in Econometrics, Academic Press, New York and

We make the following assumptions with respect to the error terms;

$$Eu_i u_{i'} = 0 \text{ for } i \neq i' \qquad Ev_t v_{t'} = 0 \text{ for } t \neq t'$$

$$Ew_{it} w_{i't} = Ew_{it'} w_{it} = Ew_{i't'} w_{it} = 0 \text{ for } i \neq i' \text{ and } t \neq t'$$

Then if we denote the number of countries by N and the number of years on which we have observations by T, we can write:

$$Euu' = \Sigma = \sigma_w^2 I_{NT} + \sigma_u^2 A + \sigma_v^2 B$$

where I_{NT} is an NT x NT identity matrix and A and B are defined as:

$$A = \begin{bmatrix} J_T & 0 & \cdots & 0 \\ 0 & J_T & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & J_T \end{bmatrix}, \qquad B = \begin{bmatrix} I_T & I_T & \cdots & I_T \\ I_T & I_T & \cdots & I_T \\ \vdots & \vdots & & \vdots \\ I_T & I_T & \cdots & I_T \end{bmatrix}$$

where J_T is a square matrix whose elements are one and A has N rows and N columns of T x T block matrices. I_T is a T x T identity matrix and B has N rows and columns of block matrices.

It is clear that OLS applied to this model is inefficient and inconsistent. To estimate this model consistently, we follow the method outlined by Wallace and Hussain.[6] We estimate σ_w^2, σ_u^2 and σ_v^2 by:

$$s_w^2 = \frac{1}{(N-1)(T-1)} \Sigma_i \Sigma_t \left(e_{it} - \frac{1}{T} e_{i.} - \frac{1}{N} e_{.t}\right)^2$$

$$s_u^2 = \frac{1}{T}\left[\sum_{i=1}^{N} \frac{e_i^2}{T(N-1)} - s_w^2\right]$$

$$s_v^2 = \frac{1}{N}\left[\sum_{t=1}^{T} \frac{e_{.t}^2}{N(T-1)} - s_w^2\right]$$

London 1974, Ch. 5, pp. 143-168; Hsiao, Cheng, "Statistical Inference for a Model with both Random Cross-Sectional and Time Effects," International Economic Review, vol. 15, no. 1, February, 1974, pp. 12-30; and Hsiao, Cheng, "Some Estimation Methods for a Random Coefficient Model," Econometrica, vol. 43, no. 2, (March, 1974), pp. 305-325.

[6] Wallace, T. D., and Hussain, Ashiq, "The Use of Error Component Models in Combining Cross Section with Time Series Data," Econometrica, vol. 37a, 1969, no. 1, pp. 55-72.

where e_{it}'s are residuals obtained by least square estimation of the original model.

Wallace and Hussain show that:

$$\lim_{\substack{N \to \infty \\ T \to \infty}} E(s_w^2 - \sigma_w^2)^r = \lim_{\substack{N \to \infty \\ T \to \infty}} E(s_u^2 - \sigma_u^2)^r = \lim_{\substack{N \to \infty \\ T \to \infty}} E(s_v^2 - \sigma_v^2)^r = 0$$

for $r = 1, 2, 3, \ldots$

Therefore, the above estimates are asymptotically unbiased and consistent, and all higher order moments vanish.

Once we have these estimates, then we can compute:

$$\hat{\Sigma} = s_w^2 I + s_u^2 A + s_v^2 B$$

and $\beta = (\beta_1 \ldots \beta_p)'$ can be estimated as:

$$\hat{\hat{\beta}} = (X'\hat{\Sigma}^{-1}X)^{-1}X'\hat{\Sigma}^{-1}Y$$

This two-stage estimator will have the following properties:

(i) $\hat{\hat{\beta}}$ has the multivariate normal limiting distribution as $\tilde{\beta}$ where $\tilde{\beta}$ is the Aitkin estimator:

$$\tilde{\beta} = (X'\Sigma^{-1}X)^{-1}X'\Sigma^{-1}Y$$

with known Σ.

(ii) $E(\hat{\hat{\beta}})$ and $\text{cov}(\hat{\hat{\beta}})$ tend to the same set of moments as those in the limiting distribution.

(iii) compared to OLS, the estimates of $\hat{\hat{\beta}}$ are asymptotically more efficient. OLS estimates are very inefficient and have unbounded asymptotic variances.

So far so good. But looking back at $\hat{\hat{\beta}}$ estimator we find out that there is an operational difficulty with this estimator. It involves inverting the matrix Σ. But this matrix is NT by NT and it is neither diagonal nor has any other implifying characteristic. NT should be very large. If it is larger than, ay, 180, usual methods of matrix inversion like Gass-Jordan method will put reat strain on the computer and as it is increased beyond 200, these methods an no more be used and we should resort to methods utilizing the expansion f the inverse of a matrix, the accuracy of which should be checked in every tep and again great inefficiencies will result. This leads us to an important ontribution by Marc Nerlove [7] which solves this problem.

We can write:

$$\Sigma = \sigma_w^2 I_{NT} + \sigma_u^2 (I_N \otimes e_T e'_T) + \sigma_v^2 (e_N e'_N \otimes I_T)$$

here e_N and e_T are vectors of orders N x 1 and T x 1 with all their elements qual to one. With the aid of a theorem originally suggested by E. J. Hannan, we an find the characteristic values of matrix Σ and diagolize it. The theorem s the following.

Theorem: Let e_N and e_T be defined as above and let ϕ_j, j = 1,...,N-1, e N-1 vectors, each N x 1 which are orthonormal and orthogonal to e_N:

$$e'_N \phi_j = 0$$

$$\phi'_j \phi_{j'} = \begin{cases} 1, & j = j' \\ 0, & j \neq j' \end{cases} \qquad (j = 1,\ldots,N-1)$$

nd let ψ_k, k = 1,...T-1, be T-1 vectors each T x 1, which are orthonormal and ach orthogonal to e_T:

$$e'_T \psi_k = 0$$

$$\psi'_k \psi_{k'} = \begin{cases} 1, & k = k' \\ 0, & k \neq k' \end{cases} \qquad (k = 1,\ldots,T-1)$$

nd let C be the NT x NT matrix

---

[7] Nerlove, Marc, "A Note on Error Components Models," Econometrica, ol. 39, no. 2, 1971, pp. 383-396.

$$C = \begin{bmatrix} \cdots \\ (e'_N/\sqrt{N}) \otimes (e'_T/\sqrt{T}) \\ (e'_N/\sqrt{N}) \otimes \psi'_1 \\ \vdots \\ (e'_N/\sqrt{N}) \otimes \psi'_{T-1} \\ \phi'_1 \otimes (e'_T/\sqrt{T}) \\ \vdots \\ \phi'_{N-1} \otimes (e'_T/\sqrt{T}) \\ \phi'_1 \otimes \psi'_1 \\ \phi'_1 \otimes \psi'_2 \\ \vdots \\ \phi'_{N-1} \otimes \psi'_{T-1} \end{bmatrix}$$

hen $C'C = I_{NT}$ and

$$C\Sigma C' = C\{\sigma_w^2 I_{NT} + \sigma_u^2(I_N \times e_T e'_T) + \sigma_v^2(e_N e'_N \times I_T)\}C'$$

$$= \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 I_{T-1} & 0 & 0 \\ 0 & 0 & \lambda_3 I_{N-1} & 0 \\ 0 & 0 & 0 & \lambda_4 I_{(N-1)(T-1)} \end{bmatrix} = D_\lambda$$

here $\lambda_1 = \sigma_w^2 - \sigma_u^2 - \sigma_v^2 + \sigma_v^2 N + \sigma_u^2 T$, $\lambda_2 = \sigma_w^2 - \sigma_u^2 - \sigma_v^2 + \sigma_v^2 N$,

$_3 = \sigma_w^2 - \sigma_u^2 - \sigma_v^2 + \sigma_u^2 T$ and $\lambda_4 = \sigma_w^2 - \sigma_u^2 - \sigma_v^2$ are the four distinct racteristic roots of Σ.[8]

Then we know that

$$\Sigma = C' D_{\lambda} C$$

$$\Sigma^{-1} = C' D_{1/\lambda} C$$

ere

$$D_{1/\lambda} = \begin{bmatrix} 1/\lambda_1 & 0 & 0 & 0 \\ 0 & (1/\lambda_2) I_{T-1} & 0 & 0 \\ 0 & 0 & (1/\lambda_3) I_{N-1} & 0 \\ 0 & 0 & 0 & (1/\lambda_4) I_{(N-1)(T-1)} \end{bmatrix}$$

d thus

$$\Sigma^{-1} = \frac{1}{\lambda_1} \frac{e_{NT} e'_{NT}}{NT} + \frac{1}{\lambda_2} \left\{ \frac{e_N e'_N}{N} \otimes I_T - \frac{e_{NT} e'_{NT}}{NT} \right\}$$

$$+ \frac{1}{\lambda_3} \left\{ I_N \otimes \frac{e_T e'_T}{T} - \frac{e_{NT} e'_{NT}}{NT} \right\}$$

$$+ \frac{1}{\lambda_4} \left\{ I_{NT} - \frac{e_N e'_N}{N} \otimes I_T - I_N \otimes \frac{e_T e'_T}{T} + \frac{e_{NT} e'_{NT}}{NT} \right\}$$

ence,

$$\hat{\hat{\beta}} = \Big(\frac{1}{\lambda_1} \frac{X' e_{NT} e'_{NT} X}{NT} + \frac{1}{\lambda_2} \Big[X' \Big(\frac{e_N e'_N}{N} \otimes I_T \Big) X - \frac{X' e_{NT} e'_{NT} X}{NT} \Big]$$

$$+ \frac{1}{\lambda_3} \Big[X' \Big(I_N \otimes \frac{e_T e'_T}{T} \Big) X - \frac{X' e_{NT} e'_{NT} X}{NT} \Big]$$

$$+ \frac{1}{\lambda_4} \Big[X'X - X' \Big(\frac{e_N e'_N}{N} \otimes I_T \Big) X - X' \Big(I_N \otimes \frac{e_T e'_T}{T} \Big) X + \frac{X' e_{NT} e'_{NT} X}{NT} \Big] \Big)^{-1}$$

---

[8] For a proof of the theorem, see Nerlove, op. cit., pp. 386-388.

$$\{\frac{1}{\lambda_1} \frac{X'e_{NT}e'_{NT}y}{NT} + \frac{1}{\lambda_2} [X'(\frac{e_N e'_N}{N} \otimes I_T)y - \frac{X'e_{NT}e'_{NT}y}{NT}]$$

$$+ \frac{1}{\lambda_3} [X'(I_N \otimes \frac{e_T e'_T}{T})y - \frac{Xe_{NT}e'_{NT}y}{NT}]$$

$$+ \frac{1}{\lambda_4} [X'y - X'(\frac{e_N e'_N}{N} \otimes I_T)y - X'(I_N \otimes \frac{e_T e'_T}{T})y + \frac{X'e_{NT}e'_{NT}y}{NT}]\}$$

This is an operational formula for $\hat{\hat{\beta}}$ and a computer program was designed to compute it.

There is one last point to be made with respect to the estimation of $\hat{\hat{\beta}}$. Note that both N and T are fixed; in other words, we should have observations on all individuals for all periods. No missing observations are allowed; if any observation is missing, then depending which way we lose fewer observations, we have to discard either an individual or a year. The latter is not feasible if we are estimating a dynamic relationship. This confronts us with another issue, the problem of missing observations and techniques dealing with it, which we will now turn to.

Missing Observations

When confronted with the situation where a subset of observations has missing components, the simplest approach is to throw away that subset and leave the estimate on the subset of complete observations. This, however, will result in loss of efficiency compared to the case where we had complete observations for the whole set.[9] The idea behind the techniques dealing with the missing observations problem is, therefore, to use the partial information contained in observations with missing components in conjunction with complete observations in order to increase the efficiency of the estimates.

---

[9] See Kmenta, Jan, Elements of Econometrics, New York, Macmillan, 1971, pp. 336-345.

The problem of missing observations in the case of pooling cross-section nd time-series data is somewhat different. As it was mentioned above the umber of individuals in each and every period should be the same. Thus, if e have one missing component in one year for an individual, we have to disard all the observations related to that individual. This is costly in terms f efficiency. What can be done is to fill in for the missing components by ome estimates of them. To see the consequences of such a procedure let us tart with the usual model

$$y = X\beta + U$$

nd partition it in the following way:[10]

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$$

ow assume X_{22} is missing and we replace it by an estimate Z of it. The OLS stimator will be:

$$\begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \end{bmatrix} = \left(\begin{bmatrix} X'_{11} & X'_{21} \\ X'_{12} & Z' \end{bmatrix} \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & Z \end{bmatrix} \right)^{-1} \begin{bmatrix} X'_{11} & X'_{21} \\ X'_{12} & Z \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

hen

$$E(\beta) = \begin{bmatrix} E(\hat{\beta}_1) \\ E(\hat{\beta}_2) \end{bmatrix} = E \left(\begin{bmatrix} X'_{11}X_{11} + X'_{21}X_{21} & X'_{11}X_{12} + X'_{21}Z \\ X'_{12}X_{11} + Z'X_{21} & X'_{12}X_{12} + Z'Z \end{bmatrix} \right)^{-1}$$

$$\begin{bmatrix} X'_{11}X_{11} + X'_{21}X_{21} & X'_{11}X_{12} + X'_{21}X_{22} \\ X'_{12}X_{11} + Z'Z_{21} & X'_{12}X_{12} + Z'X_{22} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}$$

---

[10] Note that this is not a general treatment of missing observations; ather it is only pertinent to the case where we have complete observations n y but some of X's have missing components; the case which is relevant to ur data.

because

$$E\begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = 0$$

Now if X is constant in repeated samples, then it is obvious that $E(\beta) \neq E(\beta)$. We have bias. In fact we have traded consistency for efficiency. But if X is stochastic but $E(X_1U) = 0$, then in the above formula X_{22} is replaced by $E(X_{22})$. Note that now our estimates are conditional on observed components of X. Then the only way we can have unbiased estimates is when

$$Z = E(X_{22}).$$

The simplest estimator that may be thought to have such a property is $\overline{X}_{12}$, the mean of available observations on variables with missing components. This can be true if X_{22} comes from the same population as X_{12}. But suppose the X_1 column of X which has missing components has the following relation with time:

$$X_t = X_0 + \alpha t_u + u_i \quad t - 1, \ldots, n$$

This can be the case with, say, per capita income. Then if we replace the missing components by

$$\overline{X} = X_0 + \alpha\overline{t} \quad \text{where } \overline{t} = \sum_{t=1}^{T} t_i \,/\, T_c \qquad n_c = \text{number of complete observation}$$

Then, for example, if the last year per capita income is missing, we will definitely underestimate it and if the first component is missing, it will be overestimated. In order to avoid this problem the following relationship for the variables showing trend can be assumed:

$$X_t' = X_1(1 + \overline{\beta})^{t-1}$$

where

$$\log\,(1 + \overline{\beta}) = \frac{\log X_{Ta} - \log X_1}{T_a - 1}$$

T_a = the last year on which we have observations

Distributed Lag Models

In the cost chapter of this study we had to estimate a distributed lag model. The estimation of distributed lag models involves several important problems such as determining the number of lags and colinearity in the context of generalized least square estimators discussed above.

Several alternative estimators are available to estimate such a model. A rather comprehensive sample from the list of these estimators includes[11] Almon method, Dhrymes estimator, Maddala-Rao technique, Steiglitz-McBride technique, and spectral methods. Among all these techniques, for reasons of cost of computation, degrees of freedom, number of lags that can be estimated as well as efficiency,[12] We would prefer the Almon technique. But especially in the case of the present model the Almon technique has a very important feature. The Almon scheme only reduces the number of lags by a transformation of variables without requiring any iteration or involving any change in the error term which in the present case, as was discussed above, has a complicated nature. Thus it seems appropriate to give an account of this transformation.

Let us write the distributed lag model in the form:

$$(1) \quad Y_t = \sum_{i=0}^{n-1} w(i) X_{t-u} + u_t$$

The basic assumption here is that w(i)'s are points on a polynomial of degree q + 1. Therefore, if we know q + 2 points on this polynomial, then we know all the points on it.

---

[11] Almon, S., "The Distributed Lag Between Capital Appropriations and Expenditures," Econometrica, vol. 33, no. 1, 1965, pp. 178-196; Dhrymes, P., Distributed Lags: Problems of Estimation and Formulation, Holden-Day, Inc., San Francisco, 1971; Dhrymes, P., "On the Strong Consistency of Estimators for Certain Distributed Lag Models with Autocorrelated Errors," International Economic Review, volume 12, number 2, 1971 ; Dhrymes, P., Econometrics: Statistical Foundations and Applications, Harper & Row, New York, 1970, Dhrymes, P., "Efficient Estimation of Distributed Lags with Autocorrelated Errors," International Economic Review, vol. 10, no. 1, 1969, pp. 47-67; Dhrymes, P., Klein, L., and Steiglitz, K., "Estimation of Distributed Lags," International Economic Review, vol. 11, no. 2, 1970, pp. 235-250; and Fishman, George S., Spectral Methods in Econometrics, Harvard University Press, Cambridge, Mass., 1969.

Thus, let $w(x_0) = b_0, w(x_1) = b_1 .. w(x_{q+1}) = b_{q+1}$ then we have

)
$$w_i = \sum_{j=0}^{q+1} \phi_j(i) b_j \qquad i = 0,\dots,n-1$$

ere $\phi_j(i)$'s are the values of the Lagrangian interpolation polynomials

$$\phi_0(x) = \frac{(x-x_1)(x-x_2)\quad\dots(x-x_{q+1})}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_{q+1})}$$

$$\phi_1(x) = \frac{(x-x_0)(x-x_2)\quad\dots(x-x_{q+1})}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_{q+1})}$$

$$\vdots$$

$$\phi_{q+1}(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_q)}{(x_{q+1}-x_0)(x_{q+1}-x_1)\dots(x_{q+1}-x_q)}$$

te that since we want the weights at both ends of the distributed lag to be ro, we have $b_0 = b_{q+1} = 0$, and therefore,

$$w(i) = \sum_{j=1}^{q} \phi_j(i) b_j$$

bstituting this equation in equation (1) above, we get

$$Y_t = \sum_{i=0}^{n-1} \Big(\sum_{j=1}^{q} \phi_j(i) b_j \Big) X_{t-i} = \sum_{j=1}^{q} b_j \sum_{i=0}^{n-1} \phi_j(i) X_{t-i}$$

w we define

$$Z_{tj} = \sum_{i=0}^{n-1} \phi_j(i) X_{t-i} \qquad \begin{array}{l} \text{for all } t \geq n \\ \text{and} \\ j=1,\dots q \end{array}$$

d we rewrite our model as

$$Y_t = \sum_{j=1}^{q} b_j Z_{tj} + u_t$$

---

[12] On this point see Chetty, V. K., "Estimation of Solow's Distributed Lag dels," *Econometrica*, vol. 39, no. 1, 1971, pp. 99-118.

his is a much simpler model to estimate in that the number of lagged independent ariables is reduced so we can use OLS to estimate the b_j's and then from quation (2) above we can compute the s(i)'s.

The standard errors of the w(i)'s are:

$$S_i^2 = \phi_i \sigma^2 (Z'Z)^{-1} \phi_i'$$

here

$$\phi_i = (\phi_1(i), \ldots, \phi_q(i))$$

$\sigma^2 (Z'Z)^{-1}$ is the covariance matrix of b_j's with Z being the observation atrix on A_{tj}'s. As can be seen, the virtue of this model is that it is simple nd leaves the error term intact.

e Inconsistency of OLS when Applied to an Autoregressive Form

If we want to estimate an autoregressive equation like the one in Chapter hree OLS will result in bias and inconsistent estimates. The proof of this roposition is as follows. Let the equation be:

$$y_t = \lambda y_{t-1} + \gamma X_t + u_t - \lambda u_{t-1}$$

$$= \lambda y_{t-1} + \gamma X_t + v_t$$

ere $v_t = u_t - \lambda u_{t-1}$

t

$$y = \begin{bmatrix} y_1 \\ \cdot \\ \cdot \\ \cdot \\ y_T \end{bmatrix} \qquad y_{-1} = \begin{bmatrix} y_0 \\ \cdot \\ \cdot \\ \cdot \\ y_{T-1} \end{bmatrix} \qquad X = \begin{bmatrix} x_1 \\ \cdot \\ \cdot \\ \cdot \\ x_T \end{bmatrix} \qquad v = \begin{bmatrix} v_1 \\ \cdot \\ \cdot \\ \cdot \\ v_T \end{bmatrix}$$

e OLS estimator of $\lambda + \gamma$ will be:

$$\begin{bmatrix} \hat{\lambda} \\ \hat{\gamma} \end{bmatrix} = \begin{bmatrix} y_{-1}' y_{-1} & y_{-1}' X \\ X' y_{-1} & X'X \end{bmatrix}^{-1} \begin{bmatrix} y_{-1}' y \\ x' y \end{bmatrix}$$

$$= \begin{bmatrix} y'_{-1}\,y_{-1} & y'_{-1}\,X \\ X'y_{-1} & X'X \end{bmatrix}^{-1} \left\{ \begin{bmatrix} y'_{-1}\,y_{-1} & y'_{-1}\,X \\ X'y_{-1} & X'X \end{bmatrix} \begin{bmatrix} \lambda \\ \gamma \end{bmatrix} + \begin{bmatrix} y'_{-1}\,v \\ X'\,v \end{bmatrix} \right\}$$

efore

$$E \begin{bmatrix} \hat{\lambda} \\ \hat{\gamma} \end{bmatrix} = \begin{bmatrix} \lambda \\ \gamma \end{bmatrix} + E \left\{ \begin{bmatrix} y'_{-1}\,y_{-1} & y'_{-1}\,X \\ X'y_{-1} & X'X \end{bmatrix}^{-1} \begin{bmatrix} y'_{-1}\,v \\ X'\,v \end{bmatrix} \right\}$$

LS to be unbiased the expectation of the second term should be zero. In with probability of one the necessary and sufficient condition for this to rue is that

$$E \begin{bmatrix} y'_{-1}\,v \\ X'\,v \end{bmatrix} = 0$$

the probabilistic nature of our statement.

this will not be the case because:

$$D(y'_{-1}\,v) = E\,[\sum_{t=1}^{T} y_{t-1}\,v_t] = E\,[\sum_{t=1}^{T} y_{t-1}\,u_t - \lambda \sum_{t=1}^{T} y_{t-1}\,u_{t-1}]$$

$$= -\lambda \sum_{t=1}^{T} E\,[(b \sum_{i=0}^{\infty} \lambda^i\,x_{t-i} + u_{t-1})\,u_{t-1}]$$

$$= -\lambda\,T\,\sigma^2$$

e $\sigma^2 = E(u_t^2)$ t = 1, . . . T

$E(X'v) = E\,[\sum_{t=1}^{T} X_t\,v_t] = 0$

OLS is biased.

over,

$$\text{Plim} \begin{bmatrix} \lambda \\ \gamma \end{bmatrix} = \begin{bmatrix} \lambda \\ \gamma \end{bmatrix} + \text{Plim} \left\{ \begin{bmatrix} y'_{-1}\,y_{-1} & y'_{-1}\,X \\ X'\,y_{-1} & X'\,X \end{bmatrix} \begin{bmatrix} y'_{-1}\,v \\ X'\,v \end{bmatrix} \right\}$$

$$= \begin{bmatrix} \lambda \\ \gamma \end{bmatrix} + \text{Plim} \begin{bmatrix} \frac{y'_{-1}\,y_{-1}}{T} & \frac{y'_{-1}\,X}{T} \\ \frac{X'\,y_{-1}}{T} & \frac{X'\,X}{T} \end{bmatrix}^{-1} \text{Plim} \begin{bmatrix} \frac{y'_{-1}\,v}{T} \\ \frac{X'\,v}{T} \end{bmatrix}$$

$$= \begin{bmatrix} \lambda \\ \\ \gamma \end{bmatrix} + G^{-1} \quad \text{Plim} \begin{bmatrix} \frac{y'_{-1} v}{T} \\ \\ \frac{X' v}{T} \end{bmatrix}$$

$$\text{where } G^{-1} = \text{Plim} \begin{bmatrix} \frac{y'_{-1} y_{-1}}{T} & \frac{y'_{-1} X}{T} \\ \\ \frac{X' y_{-1}}{T} & \frac{X' X}{T} \end{bmatrix}^{-1} .$$

and by assumption G^{-1} is a constant matrix. On the other hand it can be shown that:

$$E \left(\frac{X' v}{T} \right) = 0$$

and $\text{Var} \left(\frac{X' v}{T} \right) = 0$

but

$$E \left(\frac{y'_{-1} v}{T} \right) = - \lambda \sigma^2$$

and

$$\text{Var} \left(\frac{y'_{-1} v}{T} \right) = 0$$

Therefore:

$$\lim_{T \to \infty} E \left(\frac{X' v}{T} \right) = \lim_{T \to \infty} \text{Var} \left(\frac{X' v}{T} \right) = \lim_{T \to \infty} \text{Var} \left(\frac{y'_{-1} v}{T} \right) = 0$$

and

$$\lim E \left(\frac{y'_{-1} v}{T} \right) = -\lambda \quad \sigma^2$$

$$\text{Thus} \quad \text{Plim} \begin{bmatrix} \frac{y'_{-1} v}{T} \\ \\ \frac{X' v}{T} \end{bmatrix} = \begin{bmatrix} - \lambda\sigma^2 \\ \\ 0 \end{bmatrix} .$$

and therefore OLS is biased and inconsistent. Note that this would not have been the case if the error terms in our equation were independent. In our equation the error terms are not independent, that is:

$$E(v_t v_s) = E[u_t - \lambda u_{t-1}, u_s - \lambda u_{s-1}]$$

$$= E[u_t u_s - \lambda u_{t-1} u_s - \lambda u_t u_{s-1} + \lambda^2 u_{t-1} u_{s-1}]$$

Now

$$E(v_t v_s) = \begin{cases} {}^2 (1 + \lambda^2) & \text{if } S = t \\ -\lambda \sigma^2 & \text{if } S = t-1 \\ -\lambda \sigma^2 & \text{if } S = t+1 \\ 0 & \text{otherwise} \end{cases}$$

if v's were independent OLS would be consistent and asymptotically efficient but still will be biased in application to small samples.[13]

Search Technique for the Estimation of Autoregressive Forms

If OLS is biased and under certain assumptions inconsistent, what is the alternative? Different estimators are proposed for the purpose of consistent estimation of an autoregressive equation. The simplest among them is a search technique we used in this study.

Rewrite the equation as:

$$Z_t = y_t - \lambda_0 y_{t-1} = \gamma X_t + v_t \quad .$$

In this form the right hand side presents no complication and we can estimate γ for any Z_t (which depends on the value of λ_0). A procedure therefore suggests itself:

1. Choose an interval within which true λ may fall.
2. Divide the intervals into grids; the grids can be as small as is pleased.
3. Start with the smallest value in the interval, compute Z and estimate γ.
4. Repeat the procedure till that value of λ for which the sum of squared residuals minimum is found. This value of λ together with accompanying estimate of γ are our estimates.

---

[13] See, Johnson, J., Econometric Methods, 2nd edition, McGraw-Hill Book Company, New York, 1972, pp. 305-306.

It can be shown that if v's are normally and independently distributed, this
of λ is the maximum likelihood estimator of λ and is unbiased and con-
nt, $\hat{\gamma}$ is least squares estimator of γ and has all desirable properties.

Again under the assumption of

$$v = u_t - \lambda u_{t-1}$$

estimator is inconsistent but if v's are distributed normally and independently
this technique overcomes the small sample bias problem of the OLS estimator.

APPENDIX B

Input Deck To

Search Program "UBLSQ"

Card	Col.	Variable	Description
1	1-5	NCASE	# of cases (includes lag)
	6-10	NVAR	# of variables read
	11-15	NFMT	# format cards
2	1-80	IFMT	Variable format

Data Deck

Card	Col.	Variable	Description
3 + NCASE	1-10	BEGIN	Start value for B
	11-20	END	End value for B
	21-30	BINC	Increment for B
	31-35	NDEP	# of independent var.
	36-40	DEPVAR	Variable # of the dependent variables
4 + NCASE	1-3	INDEP(1)	Independent var. #1
	4-6	INDEP(2)	Independent var. #2
	7-9	INDEP(3)	Independent var. #3

(Repeat the last two cards for more regressions.)

The program limits are 100 cases, 20 variables. These values could be changed considerably if necessary.

```
      PROGRAM UBLSQ(INPUT,OUTPUT=512B,TAPE17=0,TAPE5=INPUT)
      DIMENSION IFMT(24),BUF(100),INDEP(20),A(21),XI(21,21),X(100,21),
     +XPX(21,21),YO(100),Y(100),XS(21,21),STDERR(21),T(21)
      DIMENSION XY(20)
      DIMENSION AS(20)
      INTEGER DEPVAR
      DATA (X(I,1),I=1,100)/100*1./
    1 READ 2,NCASE,NVAR,NFMT
    2 FORMAT(3I5)
      IF(NVAR.GT.100.OR.NCASE.GT.100)GO TO 888
      N=NFMT*8
      READ 3,(IFMT(I),I=1,N)
    3 FORMAT(8A10)
      DO 4 J=1,NCASE
      READ(5,IFMT)(BUF(I),I=1,NVAR)
      WRITE(17)(BUF(I),I=1,NVAR)
    4 CONTINUE
      JJ=1
   10 REWIND 17
      DO 6 I=1,21
      DO 6 J=1,21
    6 XPX(I,J)=0
      READ 5,BEGIN,END,BINC,NDEP,DEPVAR
    5 FORMAT(3F10.0,2I5)
      IF(EOF(5))999,8
    8 READ 9,INDEP
    9 FORMAT(20I3)
      PRINT 7,JJ
    7 FORMAT(1H1,10X,*REGRESSION NO.*I3//)
      PRINT 501,NCASE,DEPVAR,(INDEP(I),I=1,NDEP)
  501 FORMAT(/,* NO. OF CASES =*I4/* DEPENDENT VARIABLE IS*I3/* INDEPEND
     +ENT VARIABLES ARE*20I3/)
      PRINT 502,BEGIN,END,BINC
  502 FORMAT(* ITERATIONS TO BEGIN AT*F5.2,* AND END AT*F5.2,* PROCEEDIN
     +G BY*F6.4,* INCREMENTS*/)
      JCASE=NCASE-1
      READ(17)(BUF(I),I=1,NVAR)
      RLAG=BUF(DEPVAR)
      SUMSQY=0.
      DO 20 I=1,JCASE
      READ(17)(BUF(J),J=1,NVAR)
      DO 15 J=1,NDEP
   15 X(I,J+1)=BUF(INDEP(J))
      YO(I)=BUF(DEPVAR)
      SUMSQY=SUMSQY+YO(I)*YO(I)
   20 CONTINUE
      MDEP=NDEP+1
      DO 30 I=1,MDEP
      DO 30 K=1,MDEP
      DO 30 J=1,JCASE
   30 XPX(I,K)=XPX(K,I)=XPX(I,K)+X(J,I)*X(J,K)
      DO 35 I=1,MDEP
      DO 35 J=1,MDEP
   35 XS(I,J)=XPX(I,J)
      [illegible]
```

```
A +-*/                                  CDC 6600 FTN V3.0-V359 OPT=1  75/06/19. 05.57.48.      PAGE    1
M UBLSQ(INPUT,OUTPUT=512B,TAPE17=0,TAPE5=INPUT)
ION IFMT(24),BUF(100),INDEP(20),A(21),XI(21,21),X(100,21),
,21),YO(100),Y(100),XS(21,21),STDERR(21),T(21)
ION XY(20)
ION AS(20)
R DEPVAR
X(I,1),I=1,100)/100*1./
,NCASE,NVAR,NFMT
(3I5)
R.GT.100.OR.NCASE.GT.100)GO TO 888
*8
,(IFMT(I),I=1,N)
(8A10)
=1,NCASE
,IFMT)(BUF(I),I=1,NVAR)
17)(BUF(I),I=1,NVAR)
UE

 17
=1,21
=1,21
J)=0
,BEGIN,END,BINC,NDEP,DEPVAR
(3F10.0,2I5)
(5))999,8
,INDEP
(20I3)
7,JJ
1H1,10X,*REGRESSION NO.*I3//)
501,NCASE,DEPVAR,(INDEP(I),I=1,NDEP)
/,* NO. OF CASES =*I4/* DEPENDENT VARIABLE IS*I3/* INDEPEND
RIABLES ARE*20I3/)
502,BEGIN,END,BINC
I* ITERATIONS TO BEGIN AT*F5.2,* AND END AT*F5.2,* PROCEEDIN
5.4,* INCREMENTS*/)
NCASE-1
7)(BUF(I),I=1,NVAR)
UF(DEPVAR)
=0.
I=1,JCASE
7)(BUF(J),J=1,NVAR)
J=1,NDEP
I)=BUF(INDEP(J))
BUF(DEPVAR)
=SUMSQY+YO(I)*YO(I)
UE
DEP+1
I=1,NDEP
K=1,NDEP
J=1,JCASE
K)=APX(K,I)=APX(I,K)+X(J,I)*X(J,K)
I=1,NDEP
J=1,NDEP
)=APX(I,J)
P=
   54 FORMAT(//* VARIANCE-COVARIANCE MATRIX*/)
      DO 55 I=1,MDEP
      PRINT 56,(XS(I,J),J=1,MDEP)
   56 FORMAT(11F11.2)
      PRINT 48
   55 CONTINUE
      CALL INVERT(XPX,MDEP,Y,0,DET,21)
      PRINT 61
   61 FORMAT(//* INVERTED MATRIX*/)
      DO 65 I=1,MDEP
      PRINT 77,(XPX(I,J),J=1,MDEP)
   77 FORMAT(11E12.5)
      PRINT 48
   65 CONTINUE
C.... CHECK FOR PROPER INVERSION
      IERR=0
      DO 40 I=1,MDEP
      DO 40 J=1,MDEP
      XI(I,J)=0.
      DO 37 K=1,MDEP
   37 XI(I,J)=XI(I,J)+XS(I,K)*XPX(K,J)
      B=XI(I,J)
      IF(I.EQ.J)GO TO 38
      IF(B.LT.-.0001.OR.B.GT..0001)45,40
   38 IF(B.LT..9999.OR.B.GT.1.0001)45,40
   45 IERR=1
   40 CONTINUE
      IF(IERR.EQ.0)GO TO 50
      PRINT 46
   46 FORMAT(///* MATRIX DID NOT INVERT PROPERLY, HERE IS THE MATRIX YOU
     ** GOT INSTEAD OF THE IDENTITY MATRIX*/)
      DO 47 I=1,MDEP
      PRINT 47,(XI(I,J),J=1,MDEP)
   47 FORMAT(11F10.4)
      PRINT 48
   48 FORMAT(/)
   49 CONTINUE
      STOP
   50 B=BMIN=BEGIN
      NIT=(END-BEGIN)/BINC+1
      SSQMIN=1.E100
      DO 500 II=1,NIT
      Y(1)=YO(1)-B*RLAG
      DO 100 I=2,JCASE
  100 Y(I)=YO(I)-B*YO(I-1)
      DO 200 I=1,MDEP
      XY(I)=0.
      DO 200 J=1,JCASE
  200 XY(I)=XY(I)+X(J,I)*Y(J)
      DO 210 I=1,MDEP
      A(I)=0.
      DO 210 J=1,MDEP
  210 A(I)=A(I)+XPX(I,J)*XY(J)
      SSQ=0.
      DO 300 I=1,JCASE
```

```
T(//* VARIANCE-COVARIANCE MATRIX*/)
 I=1,MDEP
 56,(XS(I,J),J=1,MDEP)
T(11F11.2)
 48
NUE
INVERT(XPX,MDEP,Y,0,DET,21)
 61
T(//* INVERTED MATRIX*/)
 I=1,MDEP
 77,(XPX(I,J),J=1,MDEP)
T(11E12.5)
 48
NUE
 FOR PROPER INVERSION
0
 I=1,MDEP
 J=1,MDEP
J)=0.
 K=1,MDEP
J)=XI(I,J)+XS(I,K)*XPX(K,J)
I,J)
EQ.J)GO TO 38
LT.-.0001.OR.B.GT..0001)45,40
LT..9999.OR.B.GT.1.0001)45,40
1
NUE
RK.EQ.0)GO TO 50
 46
T(///* MATRIX DID NOT INVERT PROPERLY, HERE IS THE MATRIX YOU
INSTEAD OF THE IDENTITY MATRIX*/)
 I=1,MDEP
 47,(XI(I,J),J=1,MDEP)
T(11F10.4)
 48
T(/)
NUE

R=BEGIN
END-BEGIN)/BINC+1
R=1.E100
O II=1,NIT
YO(1)-B*RLAG
O I=2,JCASE
YO(I)-B*YO(I-1)
O I=1,MDEP
=0.
O J=1,JCASE
=XY(I)+X(J,I)*Y(J)
O I=1,MDEP
0.
O J=1,MDEP
A(I)+XPX(I,J)*XY(J)
.
O I=1,JCASE
```

```
      YH=0.
      DO 290 J=1,MDEP
  290 YH=YH+A(J)*X(I,J)
      SSQ=SSQ+ABS(Y(I)-YH)**2.
  300 CONTINUE
      IF(SSQ.GE.SSQMIN)GO TO 500
      SSQMIN=SSQ
      BMIN=B
      DO 400 I=1,MDEP
  400 AS(I)=A(I)
  500 B=B+BINC
      S2=SSQMIN/FLOAT(JCASE-MDEP)
      DO  503 I=1,MDEP
      STDERR(I)=SQRT(S2*XPX(I,I))
  503 T(I)=AS(I)/STDERR(I)
      PRINT 504
  504 FORMAT(///* VARIABLE*15X*A*15X*STD ERR(A)*15X*T*/1X,8(1H-),15X,1H-
     +,15X,10(1H-),15X,1H-//)
      PRINT 505,AS(1),STDERR(1),T(1)
  505 FORMAT(* CONSTANT*,9X,E12.5,9X,E12.5,8X,E12.5)
      DO 509 I=2,MDEP
      L=I-1
      PRINT 506,INDEP(L),AS(I),STDERR(I),T(I)
  506 FORMAT(3X,I3,12X,E12.5,9X,E12.5,8X,E12.5)
  509 CONTINUE
      PRINT 507,BMIN,SSQMIN,SUMSQY
  507 FORMAT(//,* B =*F10.4/* SUM SQUARES(RESIDUAL) =*E14.7/* SUM SQUARE
     +S(C(I)) =*E14.7)
      JJ=JJ+1
      GO TO 10
  888 PRINT 889
  889 FORMAT(* ERROR IN INPUT DECK. JOB TERMINATED*)
  999 STOP
      END
```

```
+-*/

=1,MDEP
J)*X(I,J)
ABS(Y(I)-YH)**2.

E.SSQMIN)GO TO 500
SQ

=1,MDEP
I)

N/FLOAT(JCASE-MDEP)
I=1,MDEP
)=SQRT(S2*XPX(I,I))
I)/STDERR(I)
04
//* VARIABLE*15X*A*15X*STD ERR(A)*15X*T*/1X,8(1H-),15X,1H-
1H-),15X,1H-//)
05,AS(1),STDERR(1),T(1)
 CONSTANT*,9X,E12.5,9X,E12.5,8X,E12.5)
=2,MDEP

06,INDEP(L),AS(I),STDERR(I),T(I)
3X,I3,12X,E12.5,9X,E12.5,8X,E12.5)
E
07,BMIN,SSQMIN,SUMSQY
//,* B =*F10.4/* SUM SQUARES(RESIDUAL) =*E14.7/* SUM SQUARE
 =*E14.7)

D
99
* ERROR IN INPUT DECK. JOB TERMINATED*)
```

123d

```
        PROGRAM      JBLSQ      [illegible]                 CDC 6600 FTN V3.0-V359 OPT=1  75/06/19. 05.57.48.

        SYMBOLIC REFERENCE MAP

     ENTRY POINTS
     [illegible] JBLSQ

     VARIABLES      SN   TYPE              RELOCATION
     4153   A            REAL        ARRAY                   13345   AS         REAL        ARRAY
     3722   B            REAL                                 3707   BEGIN      REAL
     3711   BINC         REAL                                 3723   BMIN       REAL
     4763   BUF          REAL        ARRAY                    3677   DEPVAR     INTEGER
     3720   DET          REAL                                 3710   END        REAL
     3704   I            INTEGER                              3721   IERR       INTEGER
     3733   IFMT         INTEGER     ARRAY                    3726   II         INTEGER
     4127   INDEP        INTEGER     ARRAY                    3705   J          INTEGER
     3713   JCASE        INTEGER                              3706   JJ         INTEGER
     3717   K            INTEGER                              3732   L          INTEGER
     3716   MDEP         INTEGER                              3703   N          INTEGER
     3700   NCASE        INTEGER                              3712   NDEP       INTEGER
     3702   NFMT         INTEGER                              3724   NIT        INTEGER
     3701   NVAR         INTEGER                              3714   RLAG       REAL
     3727   SSQ          REAL                                 3725   SSQMIN     REAL
    13247   STDERR       REAL        ARRAY                    3715   SUMSQV     REAL
     3731   S2           REAL                                13274   T          REAL        ARRAY
     5071   X            REAL        ARRAY                    4200   XI         REAL        ARRAY
    11155   XPX          REAL        ARRAY                   12356   XS         REAL        ARRAY
    13321   XY           REAL        ARRAY                   12212   Y          REAL        ARRAY
     3730   YH           REAL                                12046   YO         REAL        ARRAY

     FILE NAMES          MODE
         0  INPUT         FMT           2022  OUTPUT       FMT          2556  TAPE17     UNFMT            0  TAPE5

     EXTERNALS            TYPE    ARGS
            EDF           REAL     1                                  INVERT                  6
            SQRT          REAL     1 LIBRARY

     INLINE FUNCTIONS     TYPE    ARGS
            ABS           REAL     1  INTRIN                          FLOAT      REAL    1  INTRIN

46   STATEMENT LABELS
         0   1              INACTIVE                 3527   2          FMT                       3531   3          FMT
         0   4                                       3533   5          FMT                          0   6
      3540   7          FMT                             0   8                INACTIVE            3536   9          FMT
      3754  10                                          0  15                                       0  20
         0  [illegible]                                 0  35                                       0  37
[illegible] [illegible]                              3250  40                                    3247  45
[illegible] [illegible] FMT                          3620  47          FMT                       3622  48          FMT
[illegible] [illegible]                              3307  50                                    3570  54          FMT
[illegible] [illegible]                              3575  56          FMT                       3577  61          FMT
[illegible] [illegible]                              3563  77          FMT                          0 100
[illegible] [illegible]                                 0 210                                       0 290
[illegible] [illegible]                                 0 400                                    3416 500
[illegible] [illegible] FMT                          3556 502          FMT                          0 503
[illegible] [illegible] FMT                          3635 505          FMT                       3642 506          FMT
[illegible] [illegible] FMT                             0 509                                    3510 888
```

```
  RELOCATION
ARRAY               13345  AS        REAL       ARRAY
                     3707  BEGIN     REAL
                     3723  BMIN      REAL
ARRAY                3677  DEPVAR    INTEGER
                     3710  END       REAL
                     3721  IERR      INTEGER
ARRAY                3726  II        INTEGER
ARRAY                3705  J         INTEGER
                     3706  JJ        INTEGER
                     3732  L         INTEGER
                     3703  N         INTEGER
                     3712  NDEP      INTEGER
                     3724  NIT       INTEGER
                     3714  RLAG      REAL
                     3725  SSQMIN    REAL
ARRAY                3715  SUMSQY    REAL
                    13274  T         REAL       ARRAY
ARRAY                4200  XI        REAL       ARRAY
ARRAY               12356  XS        REAL       ARRAY
ARRAY               12212  Y         REAL       ARRAY
                    12046  YO        REAL       ARRAY

  2022  OUTPUT      FMT        2556  TAPE17    UNFMT          0  TAPE5      FMT

GS
1                          INVERT              6
1 LIBRARY

GS
1  INTRIN                  FLOAT     REAL      1  INTRIN

            3527   2       FMT                       3531   3        FMT
            3533   5       FMT                          0   6
               0   8         INACTIVE                3536   9        FMT
               0  15                                    0  20
               0  35                                    0  37
            3250  40                                 3247  45
            3620  47       FMT                       3622  48        FMT
            3307  50                                 3570  54        FMT
            3575  56       FMT                       3577  61        FMT
            3603  77       FMT                          0  100
               0  210                                   0  290
               0  400                                3416  500
            3556  502      FMT                          0  503
            3635  505      FMT                       3642  506       FMT
               0  509                                3510  888
```

```
        PROGRAM       JBLSQ      R +-*/                                 CDC 6600 FTN V3.0-V359 OPT=1  75/06/19. 05.57.48.

STATEMENT LABELS
 [illegible]  404      FMT                                 3513   999

STATISTICS
  PROGRAM LENGTH   104748      4412
   BUFFER LENGTH    27018      1473
```

148

```
      3513   999
```

```
SUBROUTINE  INVERT     A +-*/                              CDC 6600 FTN V3.0-V359 OPT=1  75/06/19. 05.57.4

              SUBROUTINE INVERT (A,N,B,M,DETERM,M6  )                                   756
        C     SUBROUTINE MATINV(A,N,B,M,DETERM)                                         756
        C                                                                               756
        C     MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS           756
        C                                                                               756
        C     SUBROUTINE MATINV.                                                        756
        C                                                                               756
        C     PROGRAMMED BY BURTON S. GARBOW, ARGONNE NATIONAL LABORATORY,              756
        C     AND REPORTED IN IBM 704-709 SHARE LIBRARY AS AN F402.                     756
        C     THIS SUBROUTINE COMPUTES THE INVERSE AND DETERMINANT OF                   756
        C     MATRIX A, OF ORDER N, BY THE GAUSS-JORDAN METHOD.  A-INVERSE              756
        C     REPLACES A, AND THE DETERMINANT OF A IS PLACED IN DETERM.  IF             756
        C     M = 1 THE VECTOR B CONTAINS THE CONSTANT VECTOR WHEN MATINV IS            756
        C     CALLED, AND THIS IS REPLACED WITH THE SOLUTION VECTOR.  IF M = 0,         756
        C     NO SIMULTANEOUS EQUATION SOLUTIONS ARE CALLED FOR, AND B IS NOT           756
        C     PERTINENT, N IS NOT TO EXCEED 50.                                         756
        C     A, N, B, M, AND DETERM IN THE ARGUMENT LIST ARE DUMMY VARIABLES.          756
        C                                                                               756
        C                                                                               756
              DIMENSION IPIVOT(50), A(M6,M6), B(M6,1), INDEX(50,21), PIVOT(50)          756
              EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)                 756
        C     INITIALIZATION                                                            756
          100 DETERM=1.0                                                                756
          110 DO 120 J=1,N                                                              756
          120 IPIVOT(J)=0                                                               756
          130 DO 560 I=1,N                                                              756
        C     SEARCH FOR PIVOT ELEMENT                                                  756
          140 AMAX=0.0                                                                  756
          150 DO 240 J=1,N                                                              756
          160 IF (IPIVOT(J)-1) 170, 240, 170                                            756
          170 DO 230 K=1,N                                                              756
          180 IF (IPIVOT(K)-1) 190, 230, 680                                            756
          190 IF (ABS (AMAX)-ABS (A(J,K))) 200, 230, 230                                756
          200 IROW=J                                                                    756
          210 ICOLUM=K                                                                  756
          220 AMAX=A(J,K)                                                               756
          230 CONTINUE                                                                  756
          240 CONTINUE                                                                  756
          250 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                           756
        C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                         756
          260 IF (IROW-ICOLUM) 270, 370, 270                                            756
          270 DETERM=-DETERM                                                            756
          280 DO 310 L=1,N                                                              756
          290 SWAP=A(IROW,L)                                                            756
          300 A(IROW,L)=A(ICOLUM,L)                                                     756
          310 A(ICOLUM,L)=SWAP                                                          756
          320 IF(M) 370, 370, 330                                                       756
          330 DO 360 L=1, M                                                             756
          340 SWAP=B(IROW,L)                                                            756
          350 B(IROW,L)=B(ICOLUM,L)                                                     756
          360 B(ICOLUM,L)=SWAP                                                          756
          370 INDEX(I,1)=IROW                                                           756
          380 INDEX(I,2)=ICOLUM                                                         756
          390 PIVOT(I)=A(ICOLUM,ICOLUM)                                                 756
          400 DETERM=DETERM*PIVOT(I)                                                    756
```

```
 R +-*/                          CDC 6600 FTN V3.0-V359 OPT=1  75/06/19. 05.57.48.        PAGE     1
UTINE INVERT (A,N,B,M,DETERM,M6  )                                    756
UTINE MATINV(A,N,B,M,DETERM)                                          756
                                                                      756
X INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS            756
                                                                      756
UTINE MATINV.                                                         756
                                                                      756
AMMED BY BURTON S. GARBOW, ARGONNE NATIONAL LABORATORY,               756
EPORTED IN IBM 704-709 SHARE LIBRARY AS AN F402.                      756
SUBROUTINE COMPUTES THE INVERSE AND DETERMINANT OF                    756
IX A, OF ORDER N, BY THE GAUSS-JORDAN METHOD.  A-INVERSE              756
ACES A, AND THE DETERMINANT OF A IS PLACED IN DETERM.  IF             756
 THE VECTOR B CONTAINS THE CONSTANT VECTOR WHEN MATINV IS             756
ED, AND THIS IS REPLACED WITH THE SOLUTION VECTOR.  IF M = 0,         756
IMULTANEOUS EQUATION SOLUTIONS ARE CALLED FOR, AND B IS NOT           756
NENT, N IS NOT TO EXCEED 50.                                          756
 B, M, AND DETERM IN THE ARGUMENT LIST ARE DUMMY VARIABLES.           756
                                                                      756
                                                                      756
NSION IPIVOT(50), A(M6,M6), B(M6,1), INDEX(50,2), PIVOT(50)           756
VALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)                 756
IALIZATION                                                            756
RM=1.0                                                                756
20 J=1,N                                                              756
OT(J)=0                                                               756
50 I=1,N                                                              756
CH FOR PIVOT ELEMENT                                                  756
=0.0                                                                  756
40 J=1,N                                                              756
IPIVOT(J)-1) 170, 240, 170                                            756
30 K=1,N                                                              756
IPIVOT(K)-1) 190, 230, 680                                            756
ABS (AMAX)-ABS (A(J,K))) 200, 230, 230                                756
=J                                                                    756
UM=K                                                                  756
=A(J,K)                                                               756
INUE                                                                  756
INUE                                                                  756
OT(ICOLUM)=IPIVOT(ICOLUM)+1                                           756
RCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                         756
IROW-ICOLUM) 270, 370, 270                                            756
RM=-DETERM                                                            756
10 L=1,N                                                              756
=A(IROW,L)                                                            756
OW,L)=A(ICOLUM,L)                                                     756
OLUM,L)=SWAP                                                          756
) 370, 370, 330                                                       756
60 L=1, M                                                             756
=B(IROW,L)                                                            756
OW,L)=B(ICOLUM,L)                                                     756
OLUM,L)=SWAP                                                          756
X(I,1)=IROW                                                           756
X(I,2)=ICOLUM                                                         756
T(I)=A(ICOLUM,ICOLUM)                                                 756
RM=DETERM*PIVOT(I)                                                    756
```

```
SUBROUTINE  INVERT    R +-*/                              CDC 6600 FTN V3.0-V359 OPT=1  75/06/19. 05.57.48.
            C      DIVIDE PIVOT ROW BY PIVOT ELEMENT                                         756
               410 A(ICOLUM,ICOLUM)=1.0                                                      756
               420 DO 430 L=1,N                                                              756
               430 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)                                          756
               440 IF(M) 470, 470, 450                                                       756
               450 DO 460 L=1,M                                                              756
               460 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)                                          756
            C      REDUCE NON-PIVOT ROWS                                                     756
               470 DO 550 L1=1,N                                                             756
               480 IF(L1-ICOLUM) 490, 560, 490                                               756
               490 T=A(L1,ICOLUM)                                                            756
               500 A(L1,ICOLUM)=0.0                                                          756
               510 DO 520 L=1,N                                                              756
               520 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T                                             756
               530 IF(M) 560, 560, 540                                                       756
               540 DO 550 L=1,M                                                              756
               550 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T                                             756
               560 CONTINUE                                                                  756
            C      INTERCHANGE COLUMNS                                                       756
               570 DO 670 I=1,N                                                              756
               580 L=N+1-I                                                                   756
               590 IF (INDEX(L,1)-INDEX(L,2)) 600, 670, 600                                  756
               600 JROW=INDEX(L,1)                                                           756
               610 JCOLUM=INDEX(L,2)                                                         756
               620 DO 660 K=1,N                                                              756
               630 SWAP=A(K,JROW)                                                            756
               640 A(K,JROW)=A(K,JCOLUM)                                                     756
               650 A(K,JCOLUM)=SWAP                                                          756
               660 CONTINUE                                                                  756
               670 CONTINUE                                                                  756
               680 RETURN                                                                    756
                   END                                                                       756
```

```
OT ROW BY PIVOT FLEMENT                                   756
COLUM)=1.0                                                756
,N                                                        756
)=A(ICOLUM,L)/PIVCT(I)                                    756
 470, 450                                                 756
,M                                                        756
)=B(ICOLUM,L)/PIVOT(I)                                    756
-PIVOT RCWS                                               756
=1,N                                                      756
LUM) 490, 560, 490                                        756
OLUM)                                                     756
UM)=0.0                                                   756
1,N                                                       756
(L1,L)-A(ICOLUM,L)*T                                      756
, 560, 540                                                756
1,M                                                       756
(L1,L)-B(ICOLUM,L)*T                                      756
                                                          756
GE COLUMVS                                                756
1,N                                                       756
                                                          756
(L,1)-INDEX(L,2)) 600, 670, 600                           756
X(L,1)                                                    756
DEX(L,2)                                                  756
1,N                                                       756
JPOW)                                                     756
=A(K,JCOLUM)                                              756
UM)=SWAP                                                  756
                                                          756
                                                          756
                                                          756
                                                          756
```

```
          SUBROUTINE  INVERT     R 6-8/                          CDC 6600 FTN V3.0-V359 OPT=1  75/06/19. 05.57.48.

          SYMBOLIC REFERENCE MAP

   ENTRY POINTS
      2   INVERT

   VARIABLES        SN   TYPE               RELOCATION
       0  A              REAL          ARRAY       F.P.          255  AMAX      REAL
       0  B              REAL          ARRAY       F.P.            0  DETERM    REAL                    F.P.
     247  I              INTEGER                                 254  ICOLUM    INTEGER
     [illegible]  INDEX          INTEGER       ARRAY                         256  IPIVOT    INTEGER    ARRAY
     [illegible]  IROW           INTEGER                                 246  J         INTEGER
     [illegible]  JCOLUM         INTEGER                                 253  JROW      INTEGER
     [illegible]  K              INTEGER                                 251  L         INTEGER
     [illegible]  L1             INTEGER                                   0  M         INTEGER                 F.P.
     [illegible]                 INTEGER                   F.P.            0  N         INTEGER                 F.P.
     [illegible]  PIVOT          REAL          ARRAY                     255  SWAP      REAL
     [illegible]  T              REAL

   INLINE FUNCTIONS     TYPE     ARGS
          ABS           REAL        1  INTRIN

   STATEMENT LABELS
       0  100           INACTIVE                 0  110           INACTIVE                0  120           INACTIVE
       0  130           INACTIVE                 0  140           INACTIVE                0  150           INACTIVE
       0  160           INACTIVE                 0  170           INACTIVE                0  180           INACTIVE
       0  190           INACTIVE                 0  200           INACTIVE                0  210           INACTIVE
       0  220           INACTIVE                54  230                                  57  240
       0  250           INACTIVE                 0  260           INACTIVE                0  270           INACTIVE
       0  280           INACTIVE                 0  290           INACTIVE                0  300           INACTIVE
       0  310                                    0  320           INACTIVE                0  330           INACTIVE
       0  340           INACTIVE                 0  350           INACTIVE                0  360
     112  370                                    0  380           INACTIVE                0  390           INACTIVE
       0  400           INACTIVE                 0  410           INACTIVE                0  420           INACTIVE
       0  430                                    0  440           INACTIVE                0  450           INACTIVE
       0  460                                  145  470                                   0  480           INACTIVE
       0  490           INACTIVE                 0  500           INACTIVE                0  510           INACTIVE
       0  520                                    0  530           INACTIVE                0  540           INACTIVE
       0  550                                  202  560                                   0  570           INACTIVE
       0  580           INACTIVE                 0  590           INACTIVE                0  600           INACTIVE
       0  610           INACTIVE                 0  620           INACTIVE                0  630           INACTIVE
       0  640           INACTIVE                 0  650           INACTIVE                0  660
     233  670                                  236  680

   STATISTICS
     PROGRAM LENGTH     2534B     1372
```

V236

151

```
ON
.         255  AMAX      REAL
.           0  DETERM    REAL                      F.P.
          254  ICOLUM    INTEGER
          256  IPIVOT    INTEGER     ARRAY
          246  J         INTEGER
          253  JROW      INTEGER
          251  L         INTEGER
            0  M         INTEGER                   F.P.
.           0  N         INTEGER                   F.P.
          255  SWAP      REAL


   0  110          INACTIVE                 0  120
   0  140          INACTIVE                 0  150        INACTIVE
   0  170          INACTIVE                 0  180        INACTIVE
   0  200          INACTIVE                 0  210        INACTIVE
  54  230                                  57  240
   0  260          INACTIVE                 0  270        INACTIVE
   0  290          INACTIVE                 0  300        INACTIVE
   0  320          INACTIVE                 0  330        INACTIVE
   0  350          INACTIVE                 0  360
   0  380          INACTIVE                 0  390        INACTIVE
   0  410          INACTIVE                 0  420        INACTIVE
   0  440          INACTIVE                 0  450        INACTIVE
 145  470                                   0  480        INACTIVE
   0  500          INACTIVE                 0  510        INACTIVE
   0  530          INACTIVE                 0  540        INACTIVE
 202  560                                   0  570        INACTIVE
   0  590          INACTIVE                 0  600        INACTIVE
   0  620          INACTIVE                 0  630        INACTIVE
   0  650          INACTIVE                 0  660
 236  680
```

APPENDIX C
PROGRAM DESCRIPTION
FOR ERROR COMPONENT MODEL

I. Purpose and Implementation

Program "KAMRAN was written to compute estimates of dynamic economic relations from time series of cross sections. The calculations are based upon the fundamental work of Marc Nerlove in this area.[1] Specifically, the program estimates the vector of coefficients, β in the equation:

$$\hat{\beta} = \frac{1}{\lambda_1}\{ \frac{X'e_{NT}e'_{NT}X}{NT} + \frac{1}{\lambda_2} [X'(\frac{e_N e'_N}{N} \otimes I_T) \; X - \frac{X'e_{NT}e'_{NT}X}{NT}]$$

$$+ \frac{1}{\lambda_3} [X' \; (I_N \otimes \frac{e_T e'_T}{T})X - \frac{X'e_{NT}e'_{NT}X}{NT}]$$

$$+ \frac{1}{\lambda_4} [X'X - X' \; (\frac{e_N e'_N}{N} \otimes I_T) \; X - X \; (I_N \otimes \frac{e_T e'_T}{T}) \; X + \frac{X'e_{NT}e'_{NT}X}{NT}]\}^{-1}$$

$$\{ \frac{1}{\lambda_1} \; \frac{X'e_{NT}e'_{NT}y}{NT} + \frac{1}{\lambda_2} \; [X'(\frac{e_N e'_N}{N} \otimes I_T) \; y - \frac{X'e_{NT}e'_{NT}y}{NT}]$$

$$+ \frac{1}{\lambda_3} [X' \; (I_N \otimes \frac{e_T e'_T}{T}) \; y - \frac{Xe_{NT}e'_{NT}y}{NT}]$$

$$+ \frac{1}{\lambda_4} [X'y - X' \; (\frac{e_N e'_N}{N} \otimes I_T) \; y - X' \; (I_N \otimes \frac{e_T e'_T}{T})y + \frac{X'e_{NT}e'_{NT}y}{NT}]\}$$

This involves a multiple linear regression of γ on X to provide estimates of σ_u^2, σ_v^2, and σ_w^2. A second regression is then performed using the estimated variance-covariance matrix to perform the generalized least squares. The matrices needed above to compute β provide some computational difficulties because of their size. This is discussed in section V.

---

[1] "A Note on Error Components Models," Econometrica, Vol. XXXIX, No. 2, (March, 1971), pp. 383-396.

The program was written to be run on a Control Data Corp. (CDC) 6600 series machine operating under the KRONOS 2.1 control language. The code is compatible with CDC's Fortran Extended Compiler (FTN) which has many non-ANSI standard features. Also, several features of the program are based upon routines written locally for this machine. Implementation on a different machine is discussed in section V.

II. Program Usage

A. Input to program KAMRAN is format free. Each command consists of a key word, followed by an equals sign and a string of characters which define the action to be taken. Cards can be continued by placing an asterisk (*) in column one of the continuation card. The command cards and their usage are listed below.

Command	Usage & Example
TITLE	TITLE = Dynamic Estimates of Econ. Relations Used as page header.
IOUNIT	IOUNIT = File Name (or "Cards") Defines file from which data is to be taken.
VARS	VARS = VAR001 TO VAR020, X, Y, Z, XYZ List of variable names on data file. Must correspond to format statement. Variable names may be up to 10 characters long. Names separated by commas or spaces. The "To" convention may be used to generate a long list of names easily.

FMT	FMT = (20F6.0, 4F10.4) Variable format for data.
CASES	CASES = 1,000 Number of cases read from data file. If no "Cases" card is used, data is read to end of file.
TRANS	TRANS = $\left\{\begin{matrix}0\\1\end{matrix}\right\}$ Trans = 1 indicates that subroutine Trans (user supplied) will be called after each case is read to allow transformation of data. The data is available in common block /2/ in a 100 element real array.
MVALS	MVALS = VAR001 to VAR010, X, Y (999.99, 9999.)/XYZ, VAR011 (9999.) The MVALS defines missing values for the variables. The variable list is the same as the VARS card. A maximum of three missing values may be used inside the parenthesis separated by commas. Any number of lists may be used separated by /'s. A second MVALS card will eliminate the first.
NPNC	NPNC = 3,100 This card defines the data by number of periods (time periods) and number of cases respectively. The product should equal the number of input records on "Cases" card.
INPUT	INPUT Causes data to be input, checked for missing values, transformed if requested, and output to a binary file. This must precede the "Regression" card.

REGRESSION REGRESSION = /XYZ = F(X, Y, VAR001 to VAR020)

This specifies the regression function $\left\{\begin{matrix}B\\BR\end{matrix}\right\}$ to be estimated. "XYZ" is the dependent variable list enclosed in parenthesis. Options include 1) B; estimate of Beta for error components model, 2) BR; estimates of beta plus printout of residuals for both simple regression and error components model, 3) "Space," simple regression only.

FINISH FINISH

End of run. Program is exited.

B. Program Limits

1. Maximum of 100 variables allowed on input list.
2. Maximum of three missing values per list.
3. Maximum length of format is 330 characters.
4. Title may not extend beyond 80 characters.
5. Maximum number of variables in regression is 49.
6. Only one regression per pass of data due to nature of error components model (core requirements).

C. Missing Values

If a case contains a missing value that individual is excluded from the sample for all time periods.

III. The Program

The program KAMRAN is divided into three main sections; 1) main program and utility routines, 2) regression subprogram and 3) error components subprogram. The subroutines and their functions are listed by section below:

A. Main Program

The main program provides program overload including input and decoding of instructions and execution thereof. It also dynamically allocates core for the regression and error components analysis routines. Input and instruction decoding is handled by subroutines.

1. GETNEW -- skips current control card (including continuations) and locates pointer at beginning of new card.
2. GETNXT -- returns character string up to next delimiter [(1,) =   ] in common block /IO/ in IRET. Character count is in NC. Maximum number of characters is 10. If NSEP = 1, separators are ignored and next 10 characters are returned.
3. READ -- inputs card and generates listing. Reads one card ahead to check for continuation.
4. LSTEXP -- expands a list of the form: VARXXX to VARYYY, VX, VY, VZ VARAAA to VARZZZ to one variable per word.
5. FIND -- converts expanded list as above to variable numbers by comparing with "VARS" = list.
6. IVDEC -- decodes character string (right justified) of the form VARXXX. XXX is returned as an integer value.
7. IVENC -- encodes character string of the form VARXXX. XXX is an integer parameter.

B. Multiple Regression

Subroutines MULTREG, RKGSPLT, and LINEQ were taken from program MULTREG written by Elton F. Jackson of the Dept. of Sociology, Indiana University. This is entry number 756 in the Wrubel Computing Center (IU) program library. Major modifications were made to the program to make it compatible with the error components analysis program. It would most likely be more efficient to write a single subroutine to do the regression and set up the data for error analysis.

1. MULTREG -- sets up regression and input data. File is generated on TAPE84 containing data used in regression (unformatted FORTRAN write). This same data is then used in ERRANAL.
2. LDNEQ -- does actual regression and report generating.

C. Error Components Analysis

1. ERRANAL -- computes σ_u^2, σ_v^2, σ_w^2 and error components model. Generates own output.
2. INVERT -- generalized version of MATINV written by Burton S. Garbow, Argonne National Laboratories. Inverts matrix up to order 50.

IV. Error Components Method

A. Calculating of $\sigma_u^2, \sigma_v^2, \sigma_w^2$

Data is read from TAPE84, a binary file generated by MULTREG. The estimates are computed and the residuals (e_{it}) found.

$$\sigma_w^2 = \frac{1}{(N-1)(T-1)} \sum_i^N \sum_t^T \left(e_{it} - \frac{1}{T} e_{i.} - \frac{1}{N} e_{.t}\right)^2$$

$$\sigma_u^2 = \frac{1}{T} \left[\sum_i^N \frac{e_{i.}}{T(N-1)}^2 - \sigma_w^2\right]$$

$$\sigma_v^2 = \frac{1}{N} \left[\sum_t^T \frac{e_{.t}}{N(T-1)}^2 - \sigma_w^2\right]$$

$$e_{.t} = \sum_{i=1}^N e_{it} \qquad e_{i.} = \sum_{t=1}^T e_{it}$$

N = number of indivudals

T = number of time periods

Coefficients of estimate is in /BETA/BETAW(49)

B. $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ calculated

$$\lambda_1 = \sigma_w^2 - \sigma_u^2 - \sigma_v^2 + \sigma_v^2 N + \sigma_u^2 T$$

$$\lambda_2 = \sigma_w^2 - \sigma_u^2 - \sigma_v^2 + \sigma_v^2 N$$

$$\lambda_3 = \sigma_w^2 - \sigma_u^2 \sigma_v^2 + \sigma_v^2 T$$

$$\lambda_4 = \sigma_w^2 - \sigma_v^2 - \sigma_v^2$$

Certain matrices used in the calculation of β were used several times. Their coefficients were grouped to form a new set of coefficients A1, A2, A3, A4, for use in the actual calculation.

C. Matrices

There are certain component matrices that are computed as intermediate steps in the computation of β. Because X is an NTXM matrix, the intermediate steps could often not be held in core without seriously limiting the allowed values for N and T. The method used in computation of each is described below.

I_N = identity matrix of order N

J_N = square matrix of all 1's of order N

e_N = NX1 vector of 1's

M = number of variables

NT = number of individuals

T = number of time periods

1. $X'e_{NT}e_{NT}'X = [a_{ij}]$ stored in SVCOV

$$a_{ij} = \sum_{K=1}^{NT} x_{Ki} \sum_{n=1}^{NT} S_{nj}$$

2. $X' \left(\frac{e_N e_N'}{N} \otimes I_T\right) X = [b_{ij}]$ $\otimes$ indicates Kronecker product

$$b_{ij} = \frac{1}{N} \sum_{K=1}^{NT} X_{Kj} \sum_{n=1}^{NT} X_{ni} \; f_{nK}$$

This is the cross products of the sums of X_i for each t. Stored in SUMTXY.

3. $X' \left(I_N \otimes \frac{e_T e_T'}{T}\right) X = [C_{ij}]$

$$C_{ij} = \frac{1}{T} \sum_{K=1}^{NT} X_{Kj} \cdot \left(\sum_{n=1}^{NT} X_{ni} \, g_{nK} \right)$$

$$g_{nK} = \delta_{op} \qquad \left.\begin{array}{l} o = (n + T - 1)/T \\ p = (K + T - 1)/T \end{array}\right) \text{ truncated integer part only}$$

Sum of the cross products of sums of X_i for each individual for all periods. Stored in SVT.

4. X'X -- simple variance-covariance matrix. Stored in VCOVAR.

The same calculations were also done to form various corss products of X and Y. These were stored in the same arrays as above. Y was variable number one.

V. Notes on Conversion

Two subroutines in particular are specific to I.U.'s computer. Subroutine REPLFN replaces the filename in the File Environment Table (FET) with a new name. This could be eliminated by only allowing a particular predefined file to be used for input data. Subroutine MEMORY is a monitor request which extends the field length to the specific length. The matrices for MULTREG and ERRANAL are then set up in the extended blank common. This was done so that MULTREG and ERRANAL might both remain resident and share the same storage. An alternate solution would be to overlay MULTREG and ERRANAL and have their arrays pre-dimensioned.

```
      PROGRAM KAMRAN(INPUT=1001B,OUTPUT=1001B,TAPE80,TAPE81=INPUT,TAPE83
     +=0,TAPE84=0,TAPE85=0,TAPE6=OUTPUT)
C... ...TAPE80 = GENERAL INPUT UNIT
C..... TAPE83 = TRANSFORMED INPUT (BINARY)
C... . TAPE84 = VALID DATA FOR ERROR COMPONENTS ANALYSIS
C..... TAPE85 = RESIDUALS
      COMMON / / IEX(1)
      COMMON/IO/IN(80),IC,IRET,NC,IOV,NSEP
C .... IOV*   1=OVERFLOW , 0=NORMAL , -1=END-OF-CARD , -2=EOF
C..... NSEP*   0=NORMAL , 1=TAKE NEXT TEN CHARS IGNORING SEPERATORS
      COMMON/MREG/NAMEVAR(100),AD(100,3),FMT(33),NVAR
      COMMON/Z/T(100)
      COMMON/ERROR/IERR
      COMMON/P/NPKG,NIVPKG,LOCPKG(49),DEPVAR,XIVVECT(49)
      COMMON/S/NIVNUM,DVMAT(49),VARCOV(49,49),NIVPKGT,XSUMSQY,TITLE(8),
     +NAME(1,49)
      COMMON/BETA/BETAW(49),NTIME,NEVNT,NCASE
      INTEGER TITLE,FMT
      DIMENSION RMISS(3),ITMP(100),NMISS(100)
      DIMENSION ICOM(11)
      DATA ICOM/6RIOUNIT,5RTITLE,5RTRANS,5RCASES,4RVARS,3RFMT,10RREGRESS
     +ION,5RMVALS,6RFINISH,5RINPUT,4RNPNC/
      DATA ITRANS,ICASE,LUN,NVAR,NSEP,TITLE,NCOM,NT,NCS,NTIME,NEVNT/0,
     +100000,80,0,0,8*1H ,11,0,0,0,0/
      DATA NOMISS/1/
      DATA FMT/33*1H /
    1 CALL GETNEW
      IF(IOV.EQ.-2)GO TO 1000
    5 DO 6 I=1,NCOM
      J=I
      IF(IRET.EQ.ICOM(I))GO TO 7
    6 CONTINUE
      GO TO 888
    7 GO TO (200,300,400,500,600,700,800,900,1000,1100,1200),J
C.....CHANGE INPUT UNIT
  200 CALL GETNXT
      IF(NC.GT.7)GO TO 888
      IUNIT=IRET
      IF(IRET.NE.5RCARDS)GO TO 201
      LUN=81
      GO TO 1
  201 IRET=LCS(IRET,6*(10-NC))
C.....SYSTEM ROUTINE TO CHANGE NAME IN FILE ENVIRONMENT TABLE FROM *TAPE80* TO
C.....WHATEVER IS SPECIFIED AS INPUT UNIT
      CALL REPLFN(6LTAPE80,IRET)
      GO TO 1
C.....SAVE TITLE
  300 J=1
      NSEP=1
      IC=IC+1
  301 CALL GETNXT
      IF(IOV)303,302,888
  302 IF(J.EQ.9)GO TO 888
      TITLE(J)=IRET
      J=J+1
```

```
 R +-*/                    CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE      1
RAM KAMRAN(INPUT=1001B,OUTPUT=1001B,TAPE80,TAPE81=INPUT,TAPE83
APE84=0,TAPE85=0,TAPE6=OUTPUT)
PE80 = GENERAL INPUT UNIT
E83 = TRANSFORMED INPUT (BINARY)
E84 = VALID DATA FOR ERROR COMPONENTS ANALYSIS
E85 = RESIDUALS
ON / / IEX(1)
ON/IO/IN(80),IC,IRET,NC,ICV,NSEP
*    1=OVERFLOW , 0=NORMAL , -1=END-OF-CARD , -2=EOF
P*    0=NORMAL , 1=TAKE NEXT TEN CHARS IGNORING SEPERATORS
ON/MREG/NAMEVAR(100),AD(100,3),FMT(33),NVAR
ON/Z/T(100)
ON/ERROR/IERR
ON/P/NPKG,NIVPKG,LOCPKG(49),DEPVAR,XIVVECT(49)
ON/S/NIVNUM,DVMAT(49),VARCOV(49,49),NIVPKGT,XSUMSQY,TITLE(8),
(1,49)
ON/BETA/BETAW(49),NTIME,NEVNT,NCASE
GER TITLE,FMT
NSION RMISS(3),ITMP(100),NMISS(100)
NSION ICOM(11)
 ICOM/6RIOUNIT,5RTITLE,5RTRANS,5RCASES,4RVARS,3RFMT,10RREGRESS
5RMVALS,6RFINISH,5RINPUT,4RNPNC/
 ITRANS,ICASE,LUN,NVAR,NSEP,TITLE,NCOM,NT,NCS,NTIME,NEVNT/0,
00,80,0,0,8*1H ,11,0,0,0,0/
 NOMISS/1/
 FMT/33*1H /
 GETNEW
OV.EQ.-2)GO TO 1000
 I=1,NCOM

RET.EQ.ICOM(I))GO TO 7
INUE
O 888
O (200,300,400,500,600,700,800,900,1000,1100,1200),J
GE INPUT UNIT
 GETNXT
C.GT.7)GO TO 888
T=IRET
RET.NE.5RCARDS)GO TO 201
81
O 1
=LCS(IRET,6*(10-NC))
EM ROUTINE TO CHANGE NAME IN FILE ENVIRONMENT TABLE FROM 'TAPE80' TO
EVER IS SPECIFIED AS INPUT UNIT
 REPLFN(6LTAPE80,IRET)
O 1
 TITLE

=1
C+1
 GETNXT
CV)303,302,888
.EQ.9)GO TO 888
E(J)=IRET
1
```

```
      GO TO 301
  303 NSEP=0
      GO TO 1
C..... SET FLAG TO BRANCH TO TRANS SUBROUTINE
  400 CALL GETNXT
      DECODE(10,401,IRET)L
  401 FORMAT(9X,I1)
      ITRANS=L
      IF(L.NE.0)ITRANS=1
      GO TO 1
C..... NO. OF CASES
  500 CALL GETNXT
      IF(IOV.NE.0)GO TO 888
      L=IRET.OR.LN(1H ,NC*6)
      DECODE(10,501,L)ICASE
  501 FORMAT(I10)
      GO TO 1
C..... INPUT AND EXPAND VARIABLE LIST
  600 IVAR=0
      CALL LSTEXP(IR1,NAMEVAR,100,IVAR,IER)
      IF(IER.EQ.1)GO TO 888
      NVAR=IVAR
      GO TO 1
C..... READ IN FORMAT
  700 J=1
      NSEP=1
      IC=IC+1
  701 CALL GETNXT
      IF(IOV)703,702,888
  702 FMT(J)=IRET
      J=J+1
      IF(J.EQ.34)888,701
  703 NSEP=0
      GO TO 1
C. .... SET UP MISSING VALUES LIST
  900 IF(NVAR.EQ.0)GO TO 888
      NOMISS=0
      DO 901 J=1,NVAR
  901 NMISS(J)=0
  908 CALL LSTEXP(IR1,ITMP,100,NUM,IER)
      IF(IER.EQ.1)GO TO 888
      IF(NUM.EQ.0)GO TO 1
      CALL FIND(ITMP,NUM,IER)
      IF(IER.EQ.1)GO TO 888
      J=0
  903 CALL GETNXT
      IF(IOV)888,904,888
  904 L=IRET.OR.LN(1H ,NC*6)
      J=J+1
      DECODE(10,920,L)RMISS(J)
  920 FORMAT(F10.0)
      IF(IN(IC).EQ.1R))905,902
  902 IF(J.EQ.3)888,903
  905 DO 907 I=1,NUM
      NMISS(ITMP(I))=J
```

```
301
)
 1
LAG TO BRANCH TO TRANS SUBROUTINE
ETNXT
E(10,401,IRET)L
(9X,I1)
=L
E .0)ITRANS=1
 1
F CASES
ETNXT
.NE.0)GO TO 888
.OR.LN(1H ,NC*6)
(10,501,L)ICASE
(I10)
 1
 AND EXPAND VARIABLE LIST

STEXP(1R1,NAMEVAR,100,IVAR,IER)
.EQ.1)GO TO 888
VAR
 1
IN FORMAT


 1
ETNXT
)703,702,888
=IRET

Q.34)888,701

 1
UP MISSING VALUES LIST
R.EQ.0)GO TO 888
=0
 J=1,NVAR
J)=0
STEXP(1R(,ITMP,100,NUM,IER)
.EQ.1)GO TO 888
.EQ.0)GO TO 1
IND(ITMP,NUM,IER)
.EQ.1)GO TO 888

TNXT
)888,904,888
.OR.LN(1H ,NC*6)

(10,920,L)RMISS(J)
(F10.0)
IC).EQ.1R))905,902
Q.3)888,903
 I=1,NUM
ITMP(I))=J
```

```
      DO 906 K=1,J
  906 AD(ITMP(I),K)=RMISS(K)
  907 CONTINUE
      GO TO 900
C..... END OF PROGRAM
 1000 PRINT 1001
 1001 FORMAT(//* END OF RUN.  NO CONTROL CARD ERRORS ENCOUNTERED*)
      STOP
C..... READ DATA TO BINARY FILE (TRANSFORM IF SPECIFIED)
 1100 IF(ICASE.EQ.0)ICASE=100000
      NCASE=0
      REWIND 83
      IF(LUN.EQ.80)GO TO 1104
      BACKSPACE 81
      CALL RESET
 1104 READ(LUN,FMT)(T(I),I=1,NVAR)
      IF(EOF(LUN))1150,1101
 1101 NCASE=NCASE+1
      IF(ITRANS.EQ.1)CALL TRANS
      IF(NOMISS.EQ.1)GO TO 1105
      DO 1103 I=1,NVAR
      J=NMISS(I)
      IF(J.EQ.0)GO TO 1103
      DO 1102 K=1,J
 1102 IF(T(I).EQ.AD(I,K))T(I)=99999999.
 1103 CONTINUE
 1105 WRITE(83)(T(I),I=1,NVAR)
      IF(NCASE.EQ.ICASE)1150,1104
 1150 REWIND 83
      PRINT 1151,NCASE,IUNIT
 1151 FORMAT(//,10X,I4,* CASES READ FROM*A10//)
      GO TO 1
C..... STORE THE NO. OF TIME PERIODS AND NO. OF INDIVIDUAL (NPNC)
 1200 J=1
      CALL GETNXT
      IF(IOV)1204,1201,888
 1201 L=IRET.OR.LN(1H ,NC*6)
      DECODE(10,501,L)NTIME
      CALL GETNXT
      IF(IOV)1203,1202,888
 1202 L=IRET.OR.LN(1H ,NC*6)
      DECODE(10,501,L)NEVNT
      GO TO 1
 1204 NTIME=0
 1203 NEVNT=0
      GO TO 1
C..... ERROR EXIT
  888 PRINT 889
  889 FORMAT(10(1H*),* THERE IS AN ERROR IN YOUR CONTROL CARDS AT THIS P
     *OINT.  PROCESSING CEASES. *,10(1H*))
      STOP
C.... SET UP REGRESSION LIST AND PROCESS IT
C..... REGRESSION=/VARXXX=F(VARYYY TO VARZZZ,X,Y,Z),B/
C..... B INDICATES THAT ERROR COMPONENTS ANALYSIS IS DESIRED
C.....BR INDICATES ERROR COMPNENT ANALYSIS W[illegible] PRINTOUT OF RESIDUALS
```

```
*-*/                          CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE      3
K=1,J
(I),K)=RMISS(K)
E
00
 PROGRAM
001
//* END OF RUN.  NO CONTROL CARD ERRORS ENCOUNTERED*)

ATA TO BINARY FILE (TRANSFORM IF SPECIFIED)
E.EQ.0)ICASE=100000

83
EQ.80)GO TO 1104
CE 81
SET
N,FMT)(T(I),I=1,NVAR)
LUN))1150,1101
CASE+1
NS.EQ.1)CALL TRANS
SS.EQ.1)GO TO 1105
 I=1,NVAR
(I)
.0)GO TO 1103
 K=1,J
.EQ.AD(I,K))T(I)=99999999.
E
3)(T(I),I=1,NVAR)
E.EQ.ICASE)1150,1104
83
151,NCASE,IUNIT
//,10X,I4,* CASES READ FROM*A10//)

THE NO. OF TIME PERIODS AND NO. OF INDIVIDUAL (NPNC)

TNXT
1204,1201,888
UR.LN(1H ,NC*6)
10,501,L)NTIME
TNXT
1203,1202,888
UR.LN(1H ,NC*6)
10,501,L)NEVNT




EXIT
89
10(1H*),* THERE IS AN ERROR IN YOUR CONTROL CARDS AT THIS P
PROCESSING CEASES. *,10(1H*))

REGRESSION LIST AND PROCESS IT
SION=/VARXXX=F(VARYYY TO VARZZZ,X,Y,Z),R/
CATES THAT ERROR COMPONENTS ANALUSIS IS DESIRED
CATES ERROR COMPNENT ANALYSIS W[illegible] PRINTOUT OF RESIDUALS
```

```
  800 CALL LSTEXP(IR),LOCPKG,51,NUM,IER)
      IF(IER.EQ.1.OR.LOCPKG(2).NE.1RF)GO TO 888
      DO 801 I=3,NUM
  801 LOCPKG(I-1)=LOCPKG(I)
      CALL FIND(LOCPKG,NUM-1,IER)
      J=NUM-1
      IF(IER.EQ.1)GO TO 888
      NIVPKG=NUM-2
      IB=IR=0
      CALL GETNXT
      IF(IOV)803,802,888
  802 IF(IRET.NE.1RB)GO TO 804
      IB=1
      GO TO 803
  804 IF(IRET.NE.2RBR)GO TO 888
      IB=IR=1
  803 K=100*NTIME
      IF(NTIME.EQ.0)K=100
C.... MEMORY IS DYNAMICALLY ALLOCATED TO SUBPROGRAMS 'MULTREG' AND 'ERRANAL'
C..... WITH THE USE OF THE LIBRARY PROGRAM 'MEMORY'.  'MEMORY' CHANGES THE
C..... FIELD LENGTH TO THE DESIRED ADRESS
      CALL MEMORY(LOCF(IEX(1))+K)
      REWIND 83
      REWIND 84
      CALL MULTREG(IB,IEX(1),NTIME)
      CALL MEMORY(LOCF(IEX(1)))
      IF(IB.EQ.0)GO TO 1
      IF(IERR.EQ.1)GO TO 1
C..... BEGIN ERROR COMPONENTS ANALYSIS
      K=NIVPKG+2
      J=K-1
      N=K*K
      N2=J*J
      L=NTIME*K
      IMEM=2*L+2*N+N2+1
      CALL MEMORY(LOCF(IEX(1))+IMEM)
      CALL ERRANAL(J,K,IR,NTIME,IEX(1),IEX(1+L),IEX(1+2*L),IEX(1+2*L+N),
     +IEX(1+2*L+2*N))
      CALL MEMORY(LOCF(IEX(1)))
      GO TO 1
      END
```

```
STEXP(IR),LOCPKG,S1,NUM,IER)
.EQ.1.OR.LOCPKG(2).NE.1AF)GO TO 888
 I=3,NUM
G(I-1)=LOCPKG(I)
IND(LOCPKG,NUM-1,IER)
-1
.EQ.1)GO TO 888
=NUM-2
J
EINXT
)803,802,898
T.NE.1R4)GO TO 104

803
T.NE.2RBR)GO TO 888
1
NTIME
ME.EQ.0)K=100
 IS DYNAMICALLY ALLOCATED TO SUBPROGRAMS 'MULTREG' AND 'ERRANAL'
THE USE OF THE LIBRARY PROGRAM 'MEMORY'.  'MEMORY' CHANGES THE
 LENGTH TO THE DESIRED ADRESS
EMORY(LOCF(IEX(1))+K)
 83
 84
ULTREG(IB,IEX(1),NTIME)
EMORY(LOCF(IEX(1)))
EQ.0)GO TO 1
K.EQ.1)GO TO 1
 ERROR COMPONENTS ANALYSIS
KG+2



E*K
*L+2*N+N2+1
EMORY(LOCF(IEX(1))+IMEM)
RRANAL(J,K,IR,NTIME,IEX(1),IEX(1+L),IEX(1+2*L),IEX(1+2*L+N),
2*L+2*N))
EMORY(LOCF(IEX(1)))
1
```

```
        PROGRAM     KAMRAN     P +-*/                               CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11

        SYMBOLIC REFERENCE MAP

    ENTRY POINTS
     4473  KAMRAN

    VARIABLES       SN  TYPE             RELOCATION
       144  AD          REAL       ARRAY     MREG               0  BETAW      REAL       ARRAY    BETA
        63  DEPVAR      REAL                 P                  1  DVMAT      REAL       ARRAY    S
       620  FMT         INTEGER    ARRAY     MREG            5433  I          INTEGER
      5443  IB          INTEGER                               120  IC         INTEGER             IO
      5363  ICASE       INTEGER                              5763  ICOM       INTEGER    ARRAY
      5440  IEM         INTEGER                                 0  IERR       INTEGER             ERROR
         0  IEX         INTEGER    ARRAY     / /             5447  IMEM       INTEGER
         0  IO          INTEGER    ARRAY     IO               123  IOV        INTEGER             IO
      5444  IP          INTEGER                               121  IRET       INTEGER             IO
      5453  ITMP        INTEGER    ARRAY                     5362  ITRANS     INTEGER
      5435  IUNIT       INTEGER                              5437  IVAR       INTEGER
      5434  J           INTEGER                              5442  K          INTEGER
      5436  L           INTEGER                                 2  LOCPKG     INTEGER    ARRAY    P
      5364  LUN         INTEGER                              5445  N          INTEGER
      4635  NAME        INTEGER    ARRAY     S                  0  NAMEVAR    INTEGER    ARRAY    MREG
       122  NC          INTEGER              IO                63  NCASE      INTEGER             BETA
      5365  NCOM        INTEGER                              5367  NCS        INTEGER
        62  NEVNT       INTEGER              BETA               0  NIVNUM     INTEGER             S
         1  NIVPKG      INTEGER              P               4623  NIVPKGT    INTEGER             S
      5617  NMISS       INTEGER    ARRAY                     5370  NOMISS     INTEGER
         0  NPKG        INTEGER              P                124  NSEP       INTEGER             IO
      5366  NT          INTEGER                                61  NTIME      INTEGER             BETA
      5441  NUM         INTEGER                               661  NVAR       INTEGER             MREG
      5446  N2          INTEGER                              5450  RMISS      REAL       ARRAY
         0  T           REAL       ARRAY     Z               4625  TITLE      INTEGER    ARRAY    S
        62  VARCOV      REAL       ARRAY     S                 64  XIVVECT    REAL       ARRAY    P
      4624  XSUMSQY     REAL                 S

    FILE NAMES         MODE
         0  INPUT                      1023  OUTPUT       FMT          1023  TAPE6                         2046  TAPE80
         0  TAPE81                     4070  TAPE83     UNFMT          4213  TAPE84                        4336  TAPE85

    EXTERNALS             TYPE    ARGS
            EOF         REAL        1                                   ERRANAL                   9
            FIND                    3                                   GETNEW                    0
            GETNXT                  0                                   ECS        INTEGER        2
            LN          INTEGER     2                                   LOCF       INTEGER        1
            LSTEXP                  5                                   MEMORY                    1
            MULTREG                 3                                   REPLFN                    2
            RESET                   0                                   TRANS                     0

    STATEMENT LABELS
      4474  1                                    0  5             INACTIVE                 0  6
      4513  7                                 4532  200                                 4544  201
      4553  300                               4556  301                                    0  302          INACTI
      4570  303                               4572  400                                 5371  401      FMT
      4637  500                               5373  501      FMT                        4627  600
      4637  700                               4642  701                                    0  702          INACTI
     RELOCATION
ARRAY     MREG             0  BETAW      REAL       ARRAY     BETA
          P                1  DVMAT      REAL       ARRAY     S
ARRAY     MREG          5433  I          INTEGER
                         120  IC         INTEGER              IO
                        5763  ICOM       INTEGER    ARRAY
                           0  IERR       INTEGER              ERROR
ARRAY     / /           5447  IMEM       INTEGER
ARRAY     IO             123  IOV        INTEGER              IO
                         121  IRET       INTEGER              IO
RRAY                    5362  ITRANS     INTEGER
                        5437  IVAR       INTEGER
                        5442  K          INTEGER
                           2  LOCPKG     INTEGER    ARRAY     P
                        5445  N          INTEGER
RRAY      S                0  NAMEVAR    INTEGER    ARRAY     MREG
          IO              63. NCASE      INTEGER              BETA
                        5367  NCS        INTEGER
          BETA             0  NIVNUM     INTEGER              S
       .. P             4623  NIVPKGT    INTEGER              S
RRAY                    5370  NOMISS     INTEGER
          P              124  NSEP       INTEGER              IO
                          61  NTIME      INTEGER              BETA
                         661  NVAR       INTEGER              MREG
                        5450  RMISS      REAL       ARRAY
RRAY      Z             4625  TITLE      INTEGER    ARRAY     S
ARRAY     S               64  XIVVECT    REAL       ARRAY     P
          S


   1023  CUTPUT      FMT        1023  TAPE6                  2046  TAPE80
   4070  TAPE83    UNFMT        4213  TAPE84                 4336  TAPE85

GS
1                             ERRANAL               9
3                             GETNEW                0
0                             ECS        INTEGER    2
2                             LOCF       INTEGER    1
5                             MEMORY                1
3                             REPLFN                2
0                             TRANS                 0


                0  5           INACTIVE                     0  6
             4532  200                                   4544  201
             4556  301                                      0  302         INACTIVE
             4572  400                                   5371  401     FMT
             5373  501      FMT                          4627  600
             4642  701                                      0  702         INACTIVE
```

```
        PROGRAM      KAMRAN      R +-*/                                  CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.1

  STATEMENT LABELS
   4654  703                                   5133  800                                      0  801
      0  802          INACTIVE                 5176  803                                   5171  804
   5125  888                                   5413  889     FMT                           4656  900
      0  901                                      0  902         INACTIVE                  4703  903
      0  904          INACTIVE                 4732  905                                      0  906
      0  907                                   4606  908                                   5375  920     FMT
   4753  1000                                  5377  1001    FMT                           4760  1100
      0  1101         INACTIVE                    0  1102                                  5036  1103
   4776  1104                                  5041  1105                                  5051  1150
   5406  1151    FMT                           5363  1200                                     0  1201          INACTI
      0  1202         INACTIVE                 5124  1203                                  5123  1204

  COMMON BLOCKS    LENGTH
         / /            1
         IO            85
         MREG         434
         Z            100
         ERROR          1
         P            101
         S           2510
         BETA          52

  STATISTICS
    PROGRAM LENGTH    1341B      737
     BUFFER LENGTH    4461B     2353
     COMMON LENGTH    6323B     3283
      BLANK COMMON       1B        1
```

```
                                                        0  801
  5133  800                                          5171  804
  5176  803                                          4656  900
  5413  889     FMT                                  4703  903
     0  902          INACTIVE                           0  906
  4732  905                                          5375  920     FMT
  4606  908                                          4760  1100
  5377  1001    FMT                                  5036  1103
     0  1102                                         5051  1150
  5041  1105                                            0  1201         INACTIVE
  5063  1200                                         5123  1204
  5124  1203
```

```
 737
2353
 283
   1
```

```
      SUBROUTINE GETNEW
      COMMON/IO/IN(80),IC,IRET,NC,IOV,NSEP
      DATA NSEP,IC/0,81/
      IF(IC.NE.0)GO TO 9
      IC=1
      GO TO 12
    9 IC=1
   10 CALL READ(IEOF)
      IF(IEOF.EQ.1)GO TO 99
      IF(IN(1).EQ.1R*)GO TO 10
   12 CALL GETNXT
      RETURN
   99 IOV=-2
      RETURN
      END
```

```
 h +-*/                           CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.      PAGE      1

ROUTINE GETNEW
MON/IO/IN(80),IC,IRET,NC,IOV,NSEP
A NSEP,IC/0,81/
IC.[illegible])GO TO 9

O 12

L READ(IEOF)
IEOF.EQ.1)GO TO 99
IN(1).EQ.1R*)GO TO 10
L GETNXT
URN
=-2
URN
```

```
      SUBROUTINE  GETNEW     R +-+A               CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.1

      SYMBOLIC REFERENCE MAP

 ENTRY POINTS
     1  GETNEW

 VARIABLES      SN  TYPE            RELOCATION
   120  IC          INTEGER               IO             27  IEOF      INTEGER
     0  IN          INTEGER     ARRAY     IO            123  IOV       INTEGER             IO
   121  INST        INTEGER               IO            122  NC        INTEGER             IO
   124  KSEP        INTEGER               IO

 EXTERNALS           TYPE   ARGS
        GETNXT                0                          READ                 1

 STATEMENT LABELS
     6  9                             7  10                                      17  12
    22  99

 COMMON BLOCKS   LENGTH
        IO           85

 STATISTICS
   PROGRAM LENGTH      30B      24
   COMMON LENGTH      125B      85
```

```
R                          CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.        PAGE     2
P



     RELOCATION
         IO                27  IEOF       INTEGER
ARRAY    IO               123  IOV        INTEGER                IO
         IO               122  NC         INTEGER                IO
         IO

GS
O                              READ                   1


              7  10                                         17  12




  24
  85
```

```
      SUBROUTINE GETNXT
      COMMON/IJ/IN(80),IC,IRET,NC,IOV,NSEP
      DIMENSION ISEP(6)
      DATA ISEP/1R ,1R,,1R(,1R),1R/,1R=/
      DATA IEND/0/
      IF(IEND.EQ.1)GO TO 99
      IRET=IOV=NC=IF=0
      IF(NSEP.EQ.1)GO TO 30
   10 DO 15 I=1,6
      IF(IN(IC).EQ.ISEP(I))GO TO 20
   15 CONTINUE
      GO TO 30
   20 IC=IC+1
      IF(IC.EQ.81)25,10
   25 CALL READ(IEOF)
      IF(IEOF.EQ.1)GO TO 99
      IF(IN(1).EQ.1R*)28,99
   28 IC=2
      GO TO 10
   30 IRET=SHIFT(IRET,6).OR.IN(IC)
      NC=NC+1
      IC=IC+1
      IF(IC.EQ.81)40,35
   40 CALL READ(IEOF)
      IF(IEOF.EQ.1)GO TO 95
      IF(IN(1).EQ.1R*)43,95
   43 IC=2
      GO TO 35
   35 IF(NSEP.EQ.1)31,32
   31 IF(NC.EQ.10)34,30
   32 DO 33 I=1,6
      IF(IN(IC).EQ.ISEP(I))GOTO 34
   33 CONTINUE
      IF(NC.LT.10)GO TO 30
      IOV=1
   34 RETURN
   98 IEND=1
      IC=0
      RETURN
   95 IF(NSEP.EQ.0)GO TO 98
      IRET=SHIFT(IRET,(10-NC)*6).OR.[illegible](1H ,NC*6)
      GO TO 98
   99 IOV=-1
      IC=0
      IEND=0
      RETURN
      END
```

```
R +-*/                          CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.        PAGE    1

TINE GEINXT
/IJ/IN(80),IC,IRET,NC,IDV,NSEP
SION ISEP(6)
SEP/1R ,1R.,1R(,1R),1R/,1R=/
END/0/
D.EQ.1)GO TO 99
UV=NC=IF=0
P.EQ.1)GO TO 30
I=1,6
IC).EQ.ISEP(I))GC TO 20
UE
30
1
EQ.01)25,10
EAD(IEOF)
F.EQ.1)GO TO 99
1).EQ.1R*)28,99

10
HIFT(IRET,6).OR.IN(IC)
1
1
EQ.81)40,35
EAD(IEOF)
F.EQ .1)GO TO 95
1).EQ.1R*)43,95

35
P.EQ.1)31,32
EQ,10)34,30
I=1,6
IC).EQ.ISEP(I))GOTO 34
UE
LT.10)GO TO 30


P.EQ.0)GO TO 98
HIFT(IRET,(IQ-NC)*6) .OR.IRN(IH ,NC*6)
98
```

```
        SUBROUTINE  GETNXT     R +-*/                             CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.

        SYMBOLIC REFERENCE MAP

    ENTRY POINTS
        1  GETNXT

    VARIABLES     SN  TYPE              RELOCATION
 [illegible]  I         INTEGER                                        120  IC       INTEGER               IO
 [illegible]  IEND      INTEGER                                        132  IEOF     INTEGER
 [illegible]  IF        INTEGER                                          0  IN       INTEGER    ARRAY      IO
    123  IOV       INTEGER              IO                             121  IRET     INTEGER               IO
    133  ISEP      INTEGER      ARRAY                                  122  NC       INTEGER               IO
    124  NSEP      INTEGER              IO

    EXTERNALS           TYPE   ARGS
         IRN       INTEGER   2                                          READ          1

    INLINE FUNCTIONS    TYPE   ARGS
         SHIFT      NO TYPE    2  INTRIN

    STATEMENT LABELS
      12  10                                0  15                                                  23  20
       0  25          INACTIVE              0  20          INACTIVE                                37  30
       0  31          INACTIVE             64  32                                                   0  33
     100  34                               56  35                                                   0  40          INACT
       0  43          INACTIVE            104  95                                                 101  98
     116  99

    COMMON BLOCKS   LENGTH
         IO           85

    STATISTICS
      PROGRAM LENGTH     1438     99
       COMMON LENGTH     1258     85
```

```
AP



    RELOCATION
                         120  IC         INTEGER                IO
                         132  IEOF       INTEGER
                           0  IN         INTEGER     ARRAY      IO
         IO              121  IRET       INTEGER                IO
 ARRAY                   122  NC         INTEGER                IO
         IO

RGS
 2                            READ          1

RGS
 2  INTRIN


              0   15                                   23   20
              0   28        INACTIVE                   37   30
             64   32                                    0   33
             56   35                                    0   60        INACTIVE
            104   95                                  101   98




   99
   85
```

```
      SUBROUTINE READ(IEOF)
      COMMON/IO/IN(80),IC,IRET,NC,IOV,NSEP
      DIMENSION LAST(80)
      DATA JJ/0/
      IF(JJ)2,3,5
    3 PRINT 1
    1 FORMAT(1H1)
    2 READ 10,LAST
    5 JJ=JJ+2
      DO 7 I=1,80
    7 IN(I)=LAST(I)
      IC=1
      IEOF=0
      READ 10,LAST
   10 FORMAT(80R1)
      IF(EOF(61))99,11
   11 PRINT 12,IN
   12 FORMAT(10X,80R1)
      RETURN
   99 IEOF=1
      RETURN
      ENTRY RESET
      JJ=-1
      RETURN
      END
```

```
R +-*/

UTINE READ(IEOF)
N/IC/IN(80),IC,IRET,NC,IOV,NSE
SION LAST(80)
UJ/0/
)2,3,5
 1
T(1H1)
10,LAST
+2
I=1,80
=LAST(I)

0
 10,LAST
AT(80R1)
OF(61)99,1
T 12,IN
AT(10X,80R1)
RN
=1
RN
Y RESET
1
RN
```

```
        SUBROUTINE  READ      R +-*/                   CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.

        SYMBOLIC REFERENCE MAP

    ENTRY POINTS
        4  READ          52  RESET

    VARIABLES      SN  TYPE              RELOCATION
       76  I           INTEGER                          120  IC       INTEGER              IO
        0  IEOF        INTEGER              F.P.          0  IN       INTEGER     ARRAY    IO
      123  IOV         INTEGER             IO           121  IRET     INTEGER              IO
       65  JJ          INTEGER                           77  LAST     INTEGER     ARRAY
      122  NC          INTEGER             IO           124  NSEP     INTEGER              IO

    FILE NAMES         MODE
        INPUT          FMT                 OUTPUT      FMT

    EXTERNALS           TYPE   ARGS
        EOF          REAL      1

    STATEMENT LABELS
       65  1       FMT                      12  2                                0  3        INACT
       17  5                                 0  7                               70  10     FMT
        0  11          INACTIVE             72  12      FMT                     46  99

    COMMON BLOCKS    LENGTH
        IO             85

    STATISTICS
      PROGRAM LENGTH     217B      143
       COMMON LENGTH     125B       85
```

```
 R +-*/                                  CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE      2
AP


ESET

      RELOCATION
                           120  IC          INTEGER                  10
          F.P.               0  IN          INTEGER      ARRAY       10
         10                121  IRET        INTEGER                  10
                            77  LAST        INTEGER      ARRAY
         10                124  NSEP        INTEGER                  10


         OUTPUT      FMT

GS
1



               12   2                                          0   3              INACTIVE
                0   7                                         70  10       FMT
               72  12       FMT                               46  99




 143
  85
```

```
      SUBROUTINE LSTEXP(NSP,LIST,MAX,NUM,IER)
      DIMENSION LIST(1)
      COMMON/IO/IN(80),IC,IRET,NC,IOV,NSEP
      COMMON/MREG/NAMEVAR(100),AD(100,3),FMT(33),NVAR
      LP=1
      IER=IFTO=0
   10 CALL GETNXT
      IF(IOV)88,11,99
   11 IF(IRET.EQ.2RTO)GO TO 20
      LIST(LP)=IRET
      LP=LP+1
      IF(LP.GT.MAX)GO TO 99
   15 IF(NSP.EQ.1R1)GO TO 10
      IF(IN(IC).EQ.NSP)88,10
   20 IF(NVAR.NE.0)GO TO 30
      J=IVDEC(LIST(LP-1))
      IF(J.EQ.-1)GO TO 99
      CALL GETNXT
      IF(IOV)99,21,99
   21 K=IVDEC(IRET)
      IF(K.EQ.-1.OR.K.LE.J)GO TO 99
      J=J+1
      DO 25 I=J,K
      LIST(LP)=IVENC(I)
      LP=LP+1
      IF(LP.GT.MAX)GO TO 99
   25 CONTINUE
      GO TO 15
   30 J=LIST(LP-1)
      CALL FIND(J,1,IER)
      IF(IER.EQ.1)GO TO 99
      CALL GETNXT
      IF(IOV)99,31,99
   31 K=IRET
      CALL FIND(K,1,IER)
      IF(IER.EQ.1.OR.K.LE.J)GO TO 99
      J=J+1
      DO 35 I=J,K
      LIST(LP)=NAMEVAR(I)
      LP=LP+1
      IF(LP.GT.MAX)GO TO 99
   35 CONTINUE
      GO TO 15
   99 IER=1
      RETURN
   88 NUM=LP-1
      RETURN
      END
```

```
ROUTINE LSTEXP(NSP,LIST,MAX,NUM,IER)
ENSION LIST(1)
MON/IO/IN(80),IC,IRET,NC,IOV,NSEP
MON/MAEG/NAMEVAR(100),AD(100,3),FMT(33),NVAR

LETO=0
GETNXT
OV)88,11,99
IRET.EQ.2RTO)GO TO 20
T(LP)=IRET
LP+1
LP.GT.MAX)GO TO 99
NSP.EQ.1R1)GO TO 10
IN(IC).EQ.NSP)88,10
NVAR.NE.0)GO TO 30
VDEC(LIST(LP-1))
J.EQ.-1)GO TO 99
L GETNXT
OV)99,21,99
VDEC(IRET)
K.EQ.-1.OR.K.LE.J)GO TO 99
+1
25 I=J,K
T(LP)=IVENC(I)
LP+1
LP.GT.MAX)GO TO 99
TINUE
TO 15
IST(LP-1)
L FIND(J,1,IER)
IER.EQ.1)GO TO 99
L GETNXT
OV)99,31,99
RET
L FIND(K,1,IER)
IER.EQ.1.OR.K.LE.J)GO TO 99
+1
35 I=J,K
T(LP)=NAMEVAR(I)
LP+1
LP.GT.MAX)GO TO 99
TINUE
TO 15
=1
URN
=LP-1
URN
```

```
       SUBROUTINE  LSTEXP     h +-*/                          CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.

       SYMBOLIC REFERENCE MAP

   ENTRY POINTS
       2  LSTEXP

   VARIABLES     SN   TYPE              RELOCATION
     144  AD          REAL         ARRAY     MREG           620  FMT        REAL        ARRAY     MREG
     171  I           INTEGER                               120  IC         INTEGER               IO
       0  I-I         INTEGER                F.P.           166  IFTO       INTEGER
       0  IN          INTEGER      ARRAY     IO             123  IOV        INTEGER               IO
     121  INET        INTEGER                IO             167  J          INTEGER
     170  K           INTEGER                                 0  LIST       INTEGER     ARRAY      F.P.
     165  LP          INTEGER                                 0  MAX        INTEGER                F.P.
       0  NAMEVAR     INTEGER      ARRAY     MREG           122  NC         INTEGER               IO
     124  NSFP        INTEGER                IO               0  NSP        INTEGER                F.P.
       0  NUM         INTEGER                F.P.           661  NVAR       INTEGER               MREG

   EXTERNALS             TYPE    ARGS
       FIND                       3                             GETNXT                  0
       IVDEC          INTEGER     1                             IVENC      INTEGER      1

   STATEMENT LABELS
      17  10                                0  11       INACTIVE               33  15
      42  20                                0  21       INACTIVE                0  25
      77  30                                0  31       INACTIVE                0  35
          18                              140  99

   COMMON BLOCKS    LENGTH
       IO                85
       MREG             434

   STATISTICS
       PROGRAM LENGTH     210B       136
       COMMON LENGTH     1007B       519
```

```
h +-*/                CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.      PAGE     2
P



   RELOCATION
ARRAY    MREG              620  FMT       REAL      ARRAY     MREG
                           120  IC        INTEGER             IO
          F.P.             166  IFTO      INTEGER
ARRAY    IO                123  IOV       INTEGER             IO
         IO                167  J         INTEGER
                             0  LIST      INTEGER   ARRAY      F.P.
                             0  MAX       INTEGER              F.P.
ARRAY    MREG              122  NC        INTEGER             IO
         IO                  0  NSP       INTEGER              F.P.
          F.P.             661  NVAR      INTEGER             MREG

GS
3                               GETNXT              0
1                               IVENC     INTEGER   1


                0  11             INACTIVE                 33  15
                0  21             INACTIVE                  0  25
                0  31             INACTIVE                  0  35
              140  99




 136
 519
```

```
      SUBROUTINE FIND(LIST,NUM,IER)
      COMMON/MREG/NAMEVAR(100),AD(100,3),FMT(33),NVAR
      DIMENSION LIST(1)
      IER=0
      DO 10 I=1,NUM
      J=LIST(I)
      DO 5 K=1,NVAR
      M=K
      IF(J.EQ.NAMEVAR(K))GO TO 7
    5 CONTINUE
      IER=1
      RETURN
    7 LIST(I)=M
   10 CONTINUE
      RETURN
      END
```

```
ND(LIST,NUM,IER)
MEVAR(100),AD(100,3),FMT(33),NVAR
(1)




AR(K))GO TO 7
```

```
        SUBROUTINE  FIND      R +-*/                        CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE     2

        SYMBOLIC REFERENCE MAP

  ENTRY POINTS
      2  FIND

  VARIABLES       SN  TYPE              RELOCATION
     144  AD          REAL        ARRAY     MREG             620  FMT        REAL        ARRAY     MREG
      42  I           INTEGER                                  0  IER        INTEGER                   F.P.
      43  J           INTEGER                                 44  K          INTEGER
       0  LIST        INTEGER     ARRAY      F.P.             45  M          INTEGER
       0  NAMEVAR     INTEGER     ARRAY     MREG               0  NUM        INTEGER                   F.P.
     661  NVAR        INTEGER               MREG

  STATEMENT LABELS
       0  5                             34  7                                                   0  10

  COMMON BLOCKS   LENGTH
        MREG         434

  STATISTICS
    PROGRAM LENGTH      55B       45
     COMMON LENGTH     662B      434
```

```
      FUNCTION IVDEC(IV)
C.....DECODE VARIABLE OF THE FORM 'VARXXX'
      IVDEC=-1
      IF(IRN(IV,18).NE.3RVAR)RETURN
      DECODE(10,10,IV)IVDEC
   10 FORMAT(7X,I3)
      RETURN
      END
      FUNCTION    IVDEC     K +-*/                   CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE       2

      SYMBOLIC REFERENCE MAP

   ENTRY POINTS
      IVDEC

   VARIABLES       SN   TYPE               RELOCATION
      IV              INTEGER                  F.P.            33  IVDEC       INTEGER

   EXTERNALS            TYPE    ARGS
      [illegible]       INTEGER    2

   STATEMENT LABELS
     30  10         FMT

   STATISTICS
     PROGRAM LENGTH       34B        28
```

```
      FUNCTION IVENC(IV)
C.......ENCODE VARIABLE OF THE FORM 'VARXXX'
      IVENC=-1
      IF(IV.LE.0.OR.IV .GT.999)RETURN
      ENCODE(10,10,L)IV
   10 FORMAT(I3)
      IVDEC=3RVAR
      DO 20 I=1,3
      J=LC(L,I*6).A .77B
      IF(J.EQ .1R )J=1R0
      IVDEC=LC(IVDEC,6).OR.J
   20 CONTINUE
      IVENC=IVDEC
      RETURN
      END
```

```
        FUNCTION     IVENC      h +-*/                    CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE       2

        SYMBOLIC REFERENCE MAP

   ENTRY POINTS
       2  IVENC

   VARIABLES       SN   TYPE             RELOCATION
      62  I            INTEGER                                0  IV          INTEGER                 F.P.
      61  IVDEC        INTEGER                               57  IVENC       INTEGER
      63  J            INTEGER                               60  L           INTEGER

   EXTERNALS             TYPE    ARGS
          LC           INTEGER     2

   STATEMENT LABELS
      54  10       FMT                         0  20

   STATISTICS
     PROGRAM LENGTH      64B       52
```

```
      SUBROUTINE ERRANAL(MV,NV,IRESID,NT,TIN,SVT,VCOVAR,SVCOV,RMAT)
      COMMON/MREG/NAMEVAR(100),AD(100,3),FMT(33),NVR
      COMMON/P/NPKG,NIVPKG,LOCPKG(49),DEPVAR,XIVVECT(49)
      COMMON/BETA/BETAW(49),NTIME,NEVNT,NCASE
      DIMENSION TIN(NV,NT),SVT(NT,NV),VCOVAR(NV,NV),SVCOV(NV,NV),
     $RMAT(MV,NV)
      DIMENSION SUMTXY(49),RT(25),SUMXY(49),R(25),VEC(49),BHAT(49),
     2STDERR(49),T(49)
      EQUIVALENCE (STDERR(1),SUMXY(1)),(T(1),SUMTXY(1))
      REAL L1,L2,L3,L4
C.....INITIALIZE VARIABLES
      DO 200 I=1,NTIME
      R(I)=0.
      TIN(2,I)=1.
      DO 200 J=1,NV
  200 SVT(I,J)=0.
      DO 190 I=1,NV
      BHAT(I)=SUMXY(I)=0.
      DO 190 J=1,NV
  190 SVCOV(I,J)=VCOVAR(I,J)=0.
      REWIND 84
      REWIND 85
      NVAR=NV
      NC=0
C.....READ DATA , COMPUTE RESIDULS AND SUB-MATRICES
1     DO 17 J=1,NTIME
      READ(84)(TIN(1,J),(TIN(I,J),I=3,NV))
      IF(EOF(84))30,17
17    CONTINUE
      NC=NC+1
      DO 5 I=1,NVAR
    5 SUMTXY(I)=0.
      DO 6 I=1,NTIME
    6 RT(I)=0.
      SRSD=0.
      KV=NVAR-1
      DO 20 I=1,NTIME
      DO 15 J=1,NVAR
      IF(J.LT.NVAR)RT(I)=RT(I)+BETAW(J)*TIN(J+1,I)
      SUMXY(J)=SUMXY(J)+TIN(J,I)
      SUMTXY(J)=SUMTXY(J)+TIN(J,I)
      SVT(I,J)=SVT(I,J)+TIN(J,I)
      DO 10 K=1,NVAR
   10 VCOVAR(J,K)=TIN(J,I)*TIN(K,I)+VCOVAR(J,K)
   15 CONTINUE
      RT(I)=TIN(1,I)-RT(I)
      SRSD=SRSD+RT(I)
      R(I)=R(I)+RT(I)
   20 CONTINUE
      WRITE(85)(RT(I),I=1,NTIME),SRSD
      DO 25 I=1,NVAR
      DO 25 J=1,NVAR
   25 SVCOV(I,J)=SVCOV(I,J)+SUMTXY(J)*SUMTXY(I)
   30 RN=FLOAT(NC)
      REWIND 84
      REWIND 85
      TT=FLOAT(NTIME)
      SV=0.
      DO 31 I=1,NTIME
      SV=SV+R(I)*R(I)
   31 R(I)=R(I)/RN
      SW=SU=0.
      DO 40 I=1,NC
      READ(85)(RT(J),J=1,NTIME),SRSD
      SU=SU+SRSD*SRSD
      DO 35 J=1,NTIME
      RL=RT(J)-SRSD/TT-R(J)
   35 SW=SW+RL*RL
   40 CONTINUE
      SW=SW/((RN-1.)*(TT-1.))
      SU=SU/(TT*TT*(RN-1.))-SW/TT
      SV=SV/(RN*RN*(TT-1.))-SW/RN
      S=SW+SV+SU
      RHO=SU/S
C..... CALCULATE COEFFICIENTS FOR USE IN CREATING MAIN MATRIX AND VECTOR IN
C..... ERROR COMPONENTS MODEL
      W=SV/S
      L4=(1.-RHO-W)*S
      L1=L4+(W*RN+RHO*TT)*S
      L2=L4+(W*RN)*S
      L3=L4+(RHO*TT)*S
      A1=(1./L1+1./L4-1./L2-1./L3)/(RN*TT)
      A2=(1./L2-1./L4)/RN
      A3=(1./L3-1./L4)/TT
      A4=1./L4
      DO 45 I=2,NVAR
      DO 45 J=2,NVAR
      RMAT(I-1,J-1)=SUMXY(I)*SUMXY(J)*A1+A3*SVCOV(I,J)+A4*VCOVAR(I,J)
      DO 48 K=1,NTIME
   48 RMAT(I-1,J-1)=RMAT(I-1,J-1)+A2*SVT(K,I)*SVT(K,J)
   45 CONTINUE
      DO 55 I=2,NVAR
      VEC(I-1)=SUMXY(1)*SUMXY(I)*A1+A3*SVCOV(1,I)+A4*VCOVAR(1,I)
      DO 52 J=1,NTIME
   52 VEC(I-1)=VEC(I-1)+SVT(J,1)*SVT(J,I)*A2
   55 CONTINUE
C..... SAVE 'RMAT' FOR COMPARISON AFTER INVERSION
      DO 47 I=1,KV
      DO 47 J=1,KV
   47 SVCOV(I,J)=RMAT(I,J)
      CALL INVERT(RMAT,KV,B,0,DET,KV)
      B=KV
      DO 1000 I=1,KV
      DO 1100 J=1,KV
      VCOVAR(I,J)=0.
 1100 CONTINUE
 1000 CONTINUE
```

```
          DO 1300 J=1,KV
          DO 1200 K=1,KV
          VCOVAR(I,J)=VCOVAR(I,J)+RMAT(I,K)*SVCOV(K,J)
   1200   CONTINUE
   1300   CONTINUE
   1400   CONTINUE
          DO 1500 I=1,KV
          DO 1600 J=1,KV
          IF(I.NE.J) GO TO 1700
    1800 IF(VCOVAR(I,I).LT..9999.OR.VCOVAR(I,I).GT.1.0001)GO TO 2000
          GO TO 1600
    1700 IF(VCOVAR(I,J).LT.-.0001.OR.VCOVAR(I,J).GT..0001)GO TO 2000
   1600   CONTINUE
   1500   CONTINUE
          GO TO 9000
   2000   PRINT 5000
   5000   FORMAT(*ERROR COMPONENTS MATRIX DID NOT INVERT*)
          DO 7000 I=1,KV
          WRITE(6,8000)(VCOVAR(I,J),J=1,KV)
   8000   FORMAT(1H0,11F12.5)
   7000   CONTINUE
   9000   CONTINUE
       60 DO 70 I=1,KV
          DO 70 J=1,KV
       70 BHAT(I)=BHAT(I)+VEC(J)*RMAT(I,J)
          DO 500 I=1,MV
          STDERR(I)=SQRT(RMAT(I,I))
      500 T(I)=BHAT(I)/STDERR(I)
   C..... OUTPUT BETA, STANDARD ERROR IN BETA, AND T-TEST.
          PRINT 100,RN,TT,SW,SU,SV,RHO,W,L1,L2,L3,L4
      100 FORMAT(1H1* GENERALIZED LEAST-SQUARES ESTIMATES OF B FOR ERROR COM
         +PONENTS ANALYSIS*//* NO. OF CASES = *F5.0,10X,* NO. OF PERIODS =*F
         +5.0,//* SW = *F10.4,10X,*SU = *F10.4,10X,*SV = *,F10.4/5X,*RHO = *
         +F10.4,15X,*W = *F10.4//* LAMDA(I) = *4F12.4//10X*NAME*15X*BETA*20X
         +*STD ERROR*15X*T*/10X,4(1H-),15X,4(1H-),20X,9(1H-),14X,3(1H-)//)
          PRINT 101,BHAT(1),STDERR(1),T(1)
      101 FORMAT(8X*CONSTANT*3X,F20.10,11X,F12.7,8X,F12.7)
          DO 105 I=2,KV
          L=NAMEVAR(LOCPKG(I))
          PRINT 102,L,BHAT(I),STDERR(I),T(I)
      102 FORMAT(6X,A10,3X,F20.10,11X,F12.7,8X,F12.7)
      105 CONTINUE
          PRINT 303
      303 FORMAT(5(/),* VARIANCE - COVARIANCE MATRIX*)
          DO 300 I=1,MV
          PRINT 301,(RMAT(I,J),J=1,I)
   301    FORMAT(1X,7F17.10)
          PRINT 302
   302    FORMAT (/)
   300    CONTINUE
   C..... PRINT RESIDUALS IF REQUESTED
          IF(IRESID.EQ.1)PRINT 510
      510 FORMAT(5(/),20X*LIST OF RESIDUALS*//2X*CASE*5X*PERIOD*10X*Y*10X*YH
     +8(1H-))
      SUMY=SUMR=0.
      DO 550 I=1,NC
      IF(IRESID.EQ.1)PRINT 302
      DO 520 J=1,NT
      YR=YB=0.
      READ(84)(TIN(1,1),(TIN(M,1),M=3,NV))
      DO 530 K=2,NV
      YR=YR+BETAW(K-1)*TIN(K,1)
  530 YB=YB+BHAT(K-1)*TIN(K,1)
      Y=TIN(1,2)
      RR=Y-YR
      RB=Y-YB
      SUMR=SUMR+RB*RB
      SUMY=SUMY+Y*Y
      IF(IRESID.EQ.1)PRINT 540,I,J,Y,YR,RR,YB,RB
  540 FORMAT(I5,7X,I3,7X,F10.4,6X,F10.4,12X,F11.7,11X,F10.4,10X,F11.7)
  520 CONTINUE
  550 CONTINUE
      RSQ=1.-SUMR/SUMY
      RSQADJ=1.-SUMR/SUMY*FLOAT(NC-1)/FLOAT(NC-NV+1)
      PRINT 560,RSQ,RSQADJ
  560 FORMAT(///,* R-SQUARE = *F10.6,10X,* ADJUSTED R-SQUARE = *F10.6)
      RETURN
      END
```

```
        SUBROUTINE  ERRANAL    R +-*/                              CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE      5

        SYMBOLIC REFERENCE MAP

   ENTRY POINTS
       2  ERRANAL

   VARIABLES       SN  TYPE                 RELOCATION
     144  A0           REAL        ARRAY     MREG                1274  A1          REAL
    1275  A2           REAL                                      1276  A3          REAL
    1277  A4           REAL                                      1300  B           REAL
       0  BETAW        REAL        ARRAY     BETA                1622  BHAT        REAL        ARRAY
      63  DEPVAR       REAL                  P                   1301  DET         REAL
     620  FMT          REAL        ARRAY     MREG                1254  I           INTEGER
       0  IRESID       INTEGER                F.P.               1255  J           INTEGER
    1262  K            INTEGER                                   1261  KV          INTEGER
    1302  L            INTEGER                                      2  LOCPKG      INTEGER     ARRAY     P
    1250  L1           REAL                                      1251  L2          REAL
    1252  L3           REAL                                      1253  L4          REAL
    1307  M            INTEGER                                      0  MV          INTEGER                F.P.
       0  NAMEVAR      INTEGER     ARRAY     MREG                1257  NC          INTEGEP
      63  NCASE        INTEGER               BETA                  62  NEVNT       INTEGER               BETA
       1  NIVPKG       INTEGER               P                      0  NPKG        INTEGER               P
       0  NT           INTEGER                F.P.                 61  NTIME       INTEGER               BETA
       0  NV           INTEGER                F.P.               1256  NVAR        INTEGER
     661  NVR          INTEGER               MREG                1510  R           REAL        ARRAY
    1312  RB           REAL                                      1272  RHO         REAL
    1270  RL           REAL                                         0  RHAT        REAL        ARRAY      F.P.
    1263  RN           REAL                                      1311  RR          REAL
    1313  RSQ          REAL                                      1314  RSQADJ      REAL
    1457  ST           REAL        ARRAY                         1271  S           REAL
    [illegible]  SRSD  REAL                                      1315  STDERR      REAL        ARRAY
    1267  SU           REAL                                      1304  SUMR        REAL
    1375  SUMTXY       REAL        ARRAY                         1315  SUMXY       REAL        ARRAY
    1303  SUMY         REAL                                      1265  SV          REAL
       0  SVCOV        REAL        ARRAY      F.P.                  0  SVT         REAL        ARRAY      F.P.
    1266  SW           REAL                                      1376  T           REAL        ARRAY
       0  TIN          REAL        ARRAY      F.P.               1264  TT          REAL
       0  VCOVAR       REAL        ARRAY      F.P.               1541  VEC         REAL        ARRAY
    1273  W            REAL                                        64  XIVVECT     REAL        ARRAY     P
    1310  Y            REAL                                      1306  YB          REAL
    1305  YP           REAL

   FILE NAMES          MODE
          OUTPUT         FMT                    TAPE6         FMT                    TAPF84     UNFMT                    TAPE85     UNFMT

   EXTERNALS           TYPE    ARGS
          ERF            REAL      1                                  INVERT                     6
          SQRT           REAL      1 LIBRARY

   INLINE FUNCTIONS    TYPE    ARGS
          FLOAT        REAL      1  INTRIN

   STATEMENT LABELS
      74  1                                    0  5                                           0  6
```

```
 STATEMENT LABELS
        0  20                       0  25                           234  30
        0  31                       0  35                             0  40
        0  43                       0  47                             0  48
        0  52                       0  55                             0  60          INACTIVE
        0  70                    1113  100       FMT               1153  101     FMT
 [illegible]  102       FMT           0  105                            0  190
 [illegible]  200                     0  300                         1172  301     FMT
     1175  302       FMT         1165  303       FMT                  0  500
     1177  510       FMT            0  520                            0  530
     1223  540       FMT            0  550                         1232  560     FMT
        0  1000                     0  1100                           0  1200
        0  1300                     0  1400                           0  1500
 [illegible]  1600                  540  1700                           0  1800          INACTIVE
      550  2000                  1102  5000      FMT                  0  7000
     1110  8000      FMT          573  9000

 COMMON BLOCKS    LENGTH
        MREG         434
        P            101
        SETA          52

 STATISTICS
   PROGRAM LENGTH    20148       1036
    COMMON LENGTH    11138        587
```

```
SUBROUTINE  MULTREG   R +-*/              CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE      1

               SUBROUTINE MULTREG(IB,TIN,NJ)
         C..... SUBROUTINES 'MULTREG','PKGSPLT',AND 'LINEG' WERE TAKEN FROM PROGRAM
         C.....'MULTREG' IN THE WRUBEL COMPUTING CENTER PROGRAM LIBRARY (NO. 756) AT
         C..... INDIANA UNIVERSITY. THE PROGRAM WAS WRITTEN BY ELTON F. JACKSON, DEPT.
         C.... OF SOCIOLOGY, I.U., WITH THE METHOD TAKEN FROM JOHNSTONS 'ECONOMETRICS'
               COMMON/ /IEX(1)
               COMMON/S/NIVNUM,DVMAT(1,49),VARCOV(1,49,49),NIVPKGT,XSUMSOY,TITLE(
              +8),NAME(1,49)
               COMMON/Z/T(100)
               COMMON/P/NPKG,NIVPKG(1),LOCPKG(1,49),DEPVAR(1),XIVVECT(1,49)
               COMMON/MREG/NAMEVAR(100),AD(100,3),FMT(33),NVAR
               COMMON/BETA/BETAW(49),NTIME,NEVNT,NCASE
               DIMENSION TIN(100,NJ)
               DIMENSION MODEP(1),SUMSOY(1),MDIV(1,49)
               INTEGER FMT,TITLE
               DATA NPKG/1/
               DATA (NAMEBL=8HX        )                                        756
               DATA MAXSIZE/49/                                                 3SE8SLS
               DATA IREGNO/0/
               IREGNO=IREGNO+1
         C      CONTROL CARDS ARE PRINTED OUT                                   756
               NV=NVAR
                WRITE(6,220) TITLE                                              756
         220   FORMAT (1H1,5H(MLC),10X,8A10)                                    756
         C      AT BEGINNING OF PROGRAM, REGISTERS ARE ZEROED OUT               756
         270   DO 320 I =1,NPKG                                                 756
               MXDIMIN = NIVPKG(I)                                              756
               MAXDIM = NIVPKG(I) + 1                                           756
                MODEP(I)=0                                                      756
                 SUMSOY(I)=0.0                                                  756
               DO 300 J=1,MAXDIM                                                756
                DVMAT(I,J)=0.0                                                  756
               DO 290 K=1,MAXDIM                                                756
                VARCOV(I,J,K)=0.0                                               756
           290  CONTINUE                                                        756
           300  CONTINUE                                                        756
               DO 310 J=1, MXDIMIN                                              756
                MDIV(I,J)=0                                                     756
           310  CONTINUE                                                        756
           320  CONTINUE                                                        756
         C..... BEGIN READING DATA
               KCASE=0
               NC=NCASE
               NLP=1
               IF(IB.EQ.0)GO TO 330
               NLP=NTIME
               NC=NTIME*NEVNT
               L=NIVPKG(1)+1
          ,330 IF(KCASE.EQ.NC)GO TO 340
               DO 335 I=1,NLP
               READ(83)(TIN(J,I),J=1,NVAR)
               KCASE=KCASE+1
           335 CONTINUE
               IF(IB.EQ.0)GO TO 336
```

```
      DO 332 I=1,NLP
      DO 332 J=1,L
      K=LOCPKG(I,J)
      IF(TIN(K,I).EQ.99999999.)IER=1
  332 CONTINUE
      IF(IER.EQ.1)GO TO 330
      DO 334 I=1,NLP
      DO 333 J=1,L
      K=LOCPKG(I,J)
  333 T(J)=TIN(K,I)
      WRITE(84)(T(J),J=1,L)
  334 CONTINUE
  336 DO 475 II=1,NLP
      DO 337 JJ=1,NVAR
  337 T(JJ)=TIN(JJ,II)
       CALL PKGSPLT                                                     756
C      COMPUTE THE PERSON'S CONTRIBUTION TO DVMAT AND VARCOV FOR EACH PK 756
       DO 460 I=1,NPKG                                                  756
       NOCASE = 0                                                       756
       IF (DEPVAR(I).EQ.99999999.) GO TO 370                            756
  340 LM = NIVPKG(I)+1                                                  756
      DO 360 J = 2,LM                                                   756
      IF(XIVVECT(I,J).EQ.99999999.) 350,360                             756
  350 MDIV(I,J-1)=MDIV(I,J-1) + 1                                       756
       NOCASE = 1                                                       756
  360  CONTINUE                                                         756
       IF(NOCASE.EQ. 1) 460,380                                         756
  370  MODEP(I)=MODEP(I)+1                                              756
       NOCASE = 1                                                       756
       GO TO 340                                                        756
  380 NIVNUM = NIVPKG(I) +1                                             756
       DO 410 J=1,NIVNUM                                                756
       DVMAT(I,J)=DVMAT(I,J)+XIVVECT(I,J)*DEPVAR(I)                     756
       DO 400 K=1,NIVNUM                                                756
       VARCOV(I,J,K)=VARCOV(I,J,K)+XIVVECT(I,J)*XIVVECT(I,K)            756
  400  CONTINUE                                                         756
  410  CONTINUE                                                         756
       SUMSQY(I   )=SUMSQY(I   )+DEPVAR(I)**2                           756
  460  CONTINUE                                                         756
  475 CONTINUE                                                          756
      GO TO 330
C      COMPUTATIONS ARE DONE AND PRINTED OUT                            756
  540 WRITE(6,560)IREGNO
  560 FORMAT(//* REGRESSION NO.*I3)
      I=1
       NNIVPKG=NIVPKG(I)+1                                              756
       DO 590 J=1,NNIVPKG                                               756
       KK=LOCPKG(I,J)                                                   756
      NAME(I,J)=NAMEVAR(KK)                                             756
  590  CONTINUE                                                         756
      N.IVPKG=NIVPKG(I) +2                                              756
      WRITE (6,600)(NAME(I,1),I=1,NPKG)                                 756
  600 FORMAT((1H0,8HDEP VARS,3X,11(R8,3X)))
      WRITE (6,610) (MODEP(I),I=1,NPKG)                                 756
```

218

```
          WRITE (6,611) (NAME(I,2),I=1,NPKG)                                  756
      611 FORMAT((1H0,8HIND VARS,3X,11(R8,3X)))
          WRITE(6,612) (MDIV(I,1),I=1,NPKG)                                   756
     612  FORMAT((1H ,2X,7HMD LOSS,2X,11(I3,8X)))                             756
          DO 619 K=3,NNIVPKG
          WRITE(6,613) (NAME(I,K),I=1,NPKG)                                   756
      613 FORMAT((1H ,11X,11(R8,3X)))
          WRITE(6,614) (MDIV(I,K-1),I=1,NPKG)                                 756
     614  FORMAT((1H ,11X,11(I3,8X)))                                         756
      619 CONTINUE                                                            756
      620  CONTINUE                                                           756
          I=1
          XSUMSQY = SUMSQY(I)                                                 756
           NIVNUM=NIVPKG(I) +1                                                755
          NIVPKGT = NIVPKG(I)                                                 756
          K=NIVNUM*NIVNUM
          CALL MEMORY(LOCF(IEX(NJ*100))+4*K+1)
          CALL LINEQ(1,NIVNUM,IEX(NJ+1),IEX(NJ+K+1),IEX(NJ+2*K+1),IEX(NJ+3*K
         ++1))
          CALL MEMORY(LOCF(IEX(NJ*100))+1)
                         GO TO 770                                            756
      750  WRITE (6,760)                                                      756
      760  FORMAT (23H1ERROR IN CONTROL CARDS)                                756
           STOP                                                               756
      770 RETURN
          END                                                                 756
```

```
        SUBROUTINE  MULTREG   R +-*/                         CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE       4

        SYMBOLIC REFERENCE MAP

ENTRY POINTS
    2  MULTREG

VARIABLES      SN  TYPE            RELOCATION
  144  AD          REAL        ARRAY     MREG              0  BETAW       REAL        ARRAY      BETA
   63  DEPVAR      REAL        ARRAY     P                 1  DVMAT       REAL        ARRAY      S
  620  FMT         INTEGER     ARRAY     MREG            477  I           INTEGER
    0  IB          INTEGER                F.P.           510  IER         INTEGER
    0  IEX         INTEGER     ARRAY     / /             511  II          INTEGER
  431  IREGNO      INTEGER                               502  J           INTEGER
  512  JJ          INTEGER                               503  K           INTEGER
  504  KCASE       INTEGER                               516  KK          INTEGER
  507  L           INTEGER                               514  LM          INTEGER
    2  LOCPKG      INTEGER     ARRAY     P               501  MAXDIM      INTEGER
  430  MAXSIZE     INTEGER                               520  MODEP       INTEGER     ARRAY
  522  MDIV        INTEGER     ARRAY                     517  MNIVPKG     INTEGER
  500  MXDIMIN     INTEGER                              4635  NAME        INTEGER     ARRAY      S
  427  NAMEHL      INTEGER                                 0  NAMEVAR     INTEGER     ARRAY      MREG
  505  NC          INTEGER                                63  NCASE       INTEGER                BETA
   62  NEVNT       INTEGER               BETA              0  NIVNUM      INTEGER                S
    1  NIVPKG      INTEGER     ARRAY     P              4623  NIVPKGT     INTEGER                S
    0  NJ          INTEGER                F.P.           506  NLP         INTEGER
  515  NNIVPKG     INTEGER                               513  NOCASE      INTEGER
    0  NPKG        INTEGER               P                61  NTIME       INTEGER                BETA
  476  NV          INTEGER                               661  NVAR        INTEGER                MREG
  521  SUMSQY      REAL        ARRAY                       0  T           REAL        ARRAY      Z
    0  TIN         REAL        ARRAY      F.P.          4625  TITLE       INTEGER     ARRAY      S
   62  VARCOV      REAL        ARRAY     S                64  XIVVECT     REAL        ARRAY      P
 4624  XSUMSQY     REAL                  S

FILE NAMES         MODE
      TAPE6         FMT                  TAPE83    UNFMT                 TAPE84    UNFMT

EXTERNALS            TYPE    ARGS
      LINEQ                    6                                  LOCF      INTEGER      1
      MEMORY                   1                                  PKGSPLT                0

STATEMENT LABELS
  432  220      FMT                          0  270          INACTIVE                0  290
    0  300                                   0  310                                  0  320
   67  330                                   0  332                                  0  333
    0  334                                   0  335                                154  336
    0  337                                 172  340                                  0  350          INACTIVE
  205  360                                 212  370                                216  380
    0  400                                   0  410                                245  460
    0  475                                 253  540                                435  560      FMT
    0  590                                 441  600      FMT                       445  610      FMT
  451  611      FMT                        455  612      FMT                       461  613      FMT
  464  614      FMT                          0  619                                  0  620          INACTIVE
    0  750          INACTIVE               467  760      FMT                       412  770
```

```
COMMON BLOCKS   LENGTH
      / /            1
      S           2510
      Z            100
      P            101
      MREG         434
      BETA          52

STATISTICS
  PROGRAM LENGTH     616B      398
   COMMON LENGTH   6175B     3197
    BLANK COMMON       1B        1
```

```
SUBROUTINE  PKGSPLT    K +-*/                        CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE      1

            SUBROUTINE  PKGSPLT                                                   756
      C     THIS SUBROUTINE CONSTRUCTS DV SCORE AND IV VECTOR                     756
      C                                          FOR EACH PKG FOR THAT PERSON    756
      C      FROM LOCPKG AND THE PERSON'S T VECTOR                                756
            COMMON/P/NPKG,NIVPKG(1),LOCPKG(1,49),DEPVAR(1),XIVVECT(1,49)
            COMMON/Z/T(100)
            DO 150 I=1,NPKG                                                       756
            M=LOCPKG(I,1)                                                         756
            DEPVAR(I)=T(M)                                                        756
            XIVVECT(I,1) = 1.0                                                    756
            KT = NIVPKG(I) + 1                                                    756
            DO 140 J = 2,KT                                                       756
            NP = LOCPKG(I,J)                                                      756
            XIVVECT(I,J) = T(NP)                                                  756
        140 CONTINUE                                                              756
        150 CONTINUE                                                              756
        180  RETURN                                                               756
            END                                                                   756
        SUBROUTINE  PKGSPLT   R +-*/                              CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE      2

        SYMBOLIC REFERENCE MAP

  ENTRY POINTS
      1  PKGSPLT

  VARIABLES       SN  TYPE            RELOCATION
     63  DEPVAR     REAL       ARRAY     P                 25  I           INTEGER
     30  J          INTEGER                                27  KT          INTEGER
      2  LOCPKG     INTEGER    ARRAY     P                 26  K           INTEGER
      1  NIVPKG     INTEGER    ARRAY     P                 31  NP          INTEGER
      0  NPKG       INTEGER              P                  0  T           REAL       ARRAY    Z
     64  XIVVECT    REAL       ARRAY     P

  STATEMENT LABELS
      0  140                                      0  150                                    0  180        INACTIVE

  COMMON BLOCKS   LENGTH
         P           101
         Z           100

  STATISTICS
    PROGRAM LENGTH      32B       26
     COMMON LENGTH     311B      201
```

```
      SUBROUTINE LINEQ(PKGNO,JN,VERT,IDEN,TVARCOV,ZEROR)
      COMMON/ERROR/IERR
      COMMON/S/IVNUM,DVMAT(1,49),VARCOV(1,49,49),NIVPKG,SUMSQY,TITLE(8),
     +NAMEPKG(1,49)
      COMMON/BETA/SOL(49),NTIME,NEVNT,NCASE
      DIMENSION BETAW(49)
      DIMENSION VERT(JN,JN),IDEN(JN,JN),TVARCOV(JN,JN),ZEROR(JN,JN)
      DIMENSION SOL(49),CITY(7),STDERR(49),TTEST(49),TDVMAT(49),NAMET(49
     +),XMEAN(49),STDEV(49),SIGN(49)
      REAL IDEN
      INTEGER PKGNO
      DATA MORE/0/
      IERR=0
      NUMCASE = VARCOV(PKGNO,1,1)                                       756
      GMEANY = DVMAT(PKGNO,1)/VARCOV(PKGNO,1,1)                         756
      TOTVAR = SUMSQY - VARCOV(PKGNO,1,1)*GMEANY**2                     756
      TOTVN=TOTVAR/VARCOV(PKGNO,1,1)                                    756
      YSTDEV=SQRT(TOTVN)
      WRITE(6,510) NUMCASE
  510 FORMAT(1H ,18HNUMBER OF CASES = ,I5)
      HOLDSQ=2.0                                                        756
      HOLDSS=99999999.                                                  756
      NT= IVNUM                                                         756
      DO 10 I = 1,IVNUM                                                 756
      TDVMAT(I) = DVMAT(PKGNO,I)                                        756
      NAMET(I) = NAMEPKG(PKGNO,I)                                       756
      DO 11 J = 1,IVNUM                                                 756
      TVARCOV(I,J) = VARCOV(PKGNO,I,J)                                  756
   11 CONTINUE                                                          756
   10 CONTINUE                                                          756
      GO TO 99                                                          756
   20 DO 90 LP = 2,IVNUM                                                756
      IF(LP.EQ.2) HOLDSQ = RSQ                                          756
      IF(LP.EQ.2) HOLDSS = EXPLSS                                       756
      NT = IVNUM - 1                                                    756
      LQ = LP - 1                                                       756
      DO 21 LA = 1,LQ                                                   756
      TDVMAT(LA) = DVMAT(PKGNO,LA)                                      756
      NAMET(LA) = NAMEPKG(PKGNO,LA)                                     756
      DO 22 LB = 1,IVNUM                                                756
      TVARCOV (LA,LB) = VARCOV(PKGNO,LA,LB)                             756
   22 CONTINUE                                                          756
   21 CONTINUE                                                          756
      IF(LP.GT.NT) GO TO 33                                             756
      DO 23 LA = LP,NT                                                  756
      LD = LA + 1                                                       756
      TDVMAT(LA) = DVMAT(PKGNO,LD)                                      756
      NAMET(LA) = NAMEPKG(PKGNO,LD)                                     756
      DO 24 LB = 1,IVNUM                                                756
      TVARCOV(LA,LB) = VARCOV(PKGNO,LD,LB)                              756
   24 CONTINUE                                                          756
   23 CONTINUE                                                          756
   31 DO 25 LA = 1,NT                                                   756
      DO 26 LB = LP,NT                                                  756
      LC = LB + 1                                                       756
```

```
            TVARCOV(LA,LB) = TVARCCV(LA,LC)                              756
      26    CONTINUE                                                     756
      25    CONTINUE                                                     756
      33    GO TO 99                                                     756
      32    CONTINUE                                                     756
      90    CONTINUE                                                     756
            RETURN                                                       756
         99 DO 110 I=1,NT                                                756
            SOL(I) = TDVMAT(I)                                           756
            DO 100 J=1,NT                                                756
            VERT(I,J)=TVARCOV(I,J)                                       756
            IDEN(I,J)=0.0                                                756
        100 CONTINUE                                                     756
        110 CONTINUE                                                     756
            CALL INVERT(VERT,NT,SOL,1,DETERM,JN)
      C     CHECK TO SEE IF THE MATRIX INVERTED PROPERLY                 756
            DO 140 K = 1,NT                                              756
            DO 130 L = 1,NT                                              756
            DO 120 J = 1,NT                                              756
            IDEN(K,L)=IDEN(K,L)+TVARCOV(K,J)*VERT(J,L)                   756
      120   CONTINUE                                                     756
      130   CONTINUE                                                     756
      140   CONTINUE                                                     756
            DO 1141 I = 1,NT                                             756
            DO 1142 J = 1,NT                                             756
            IF(I.NE.J) GO TO 1146                                        756
      1145  IF((IDEN(I,I).LT..9999999).OR.(IDEN(I,I).GT.1.0000001))GO TO 1150  756
            GO TO 1142                                                   756
      1146  IF((IDEN(I,J).LT.-.00001).OR.(IDEN(I,J).GT..00001))GO TO 1150     756
      1142  CONTINUE                                                     756
      1141  CONTINUE                                                     756
            GO TO 3000                                                   756
      1150  WRITE(6,1160)                                                756
      1160  FORMAT(1H1,118H ATTENTION PLEASE. YOUR MATRIX DID NOT INVERT CORRE 756
           1CTLY - HERE IS THE MATRIX YOU GOT INSTED OF THE IDENTITY MATRIX.   756
           2      //)                                                    756
            DO 1210 I = 1,NT                                             756
            WRITE(6,1220) (IDEN(I,J),J=1,NT)                             756
      1220  FORMAT(1H0,11F12.5)                                          756
      1210  CONTINUE                                                     756
            GO TO 3000
            IERR=1
            RETURN                                                       756
      C                                                                  756
      3000  IF(HOLDSQ.NE.2.0) GO TO 249                                  756
            DO 248 MP = 1,IVNUM                                          756
            SIGN(MP) = SOL (MP)                                          756
      248   CONTINUE                                                     756
      249   NDFONE=NT-1                                                  756
            NDFTWO=NUMCASE-NT                                            756
            DFTWO = NDFTWO                                               756
            BTAXY = 0.0                                                  756
            DO 250 KP=1,NT                                               756
            BTAXY=BTAXY+SOL(KP)*TDVMAT(KP)                               756
        250 CONTINUE                                                     756
      EXPLSS=BTAXY-((1/TVARCOV(1,1))*TDVMAT(1)**2)                       756
      UNEXPSS  = TOTVAR-EXPLSS                                           756
      RSQ = EXPLSS/TOTVAR                                                756
      RMULT=SQRT(RSQ)                                                    756
      EXPLVN=EXPLSS/VARCOV(PKGNO,1,1)                                    756
      UNEXPVN =UNEXPSS /VARCOV(PKGNO,1,1)                                756
      XNIVPKG= NT-1                                                      756
      FRATIO=(RSQ*(VARCOV(PKGNO,1,1)-XNIVPKG-1.0))/((1.0-RSQ)*XNIVPKG)   756
      ADDSS=HOLDSS-EXPLSS                                                756
      PARTNUM = ADDSS/UNEXPSS                                            756
      PARTSQ = HOLDSQ - RSQ                                              756
      PARTR=SQRT(PARTNUM)                                                756
      IF(HOLDSQ.EQ.2.) GO TO 159                                         756
      IF(SIGN(LP).LT.0.0) PARTR = PARTR *(-1.0)                          756
      WRITE(6,314) NAMEPKG(PKGNO,LP)                                     756
  314 FORMAT(1H0,19HPARTIAL CONTRIB OF ,A8)                              756
      WRITE(6,313) PARTR,PARTSQ,ADDSS                                    756
  313 FORMAT(1H ,10X,12HPARTIAL R = ,F7.4,10X,18HADD PROP OF VAR = ,F7.4 756
     1,10X,11HSS ADDED = ,F13.4)                                         756
      IF (MORE.EQ.1) GO TO 32                                            756
  159 DO 700 LL=2,NT                                                     756
      STDERR(LL)=SQRT((UNEXPSS /DFTWO)*VERT(LL,LL))                      756
      TTEST(LL)=SOL(LL)/STDERR(LL)                                       756
      XMEAN(LL)=TVARCOV(1,LL)/TVARCOV(1,1)                               756
      STDEV(LL)=SQRT((TVARCOV(LL,LL)/TVARCOV(1,1))-XMEAN(LL)**2)         756
      BETAW(LL)=SOL(LL)*STDEV(LL)/YSTDEV                                 756
700   CONTINUE                                                           756
      WRITE(6,160) NDFTWO                                                756
160   FORMAT (1H0,19X,9HA AND B*S,3X,14HSTD ERROR OF B, 3X,5HT(DF=,I4,1H 756
     1),5X,12HBETA WEIGHTS , 9X, 5HMEANS,  8X,8HSTD DEVS )               756
      WRITE(6,230) NAMET(1),SOL(1),GMEANY,YSTDEV                         756
  230 FORMAT(1H0,2HDV,7X,R8,2X,F8.3,54X,F7.2,8X,F8.2)
      WRITE(6,231) NAMET(2),SOL(2),STDERR(2),TTEST(2),BETAW(2),XMEAN(2), 756
     1STDEV(2)                                                           756
  231 FORMAT(1H0,3HIVS,6X,R8,2X,F8.3,4X,F10.4,9X,F8.3,9X,F8.3,6X,F7.2,8X 756
     1,F8.2)                                                             756
      DO 232 I=3,NT                                                      756
      WRITE(6,233) NAMET(I),SOL(I),STDERR(I),TTEST(I),BETAW(I),XMEAN(I), 756
     1STDEV(I)                                                           756
  233 FORMAT(1H ,9X,R8,2X,F8.3,4X,F10.4,9X,F8.3,9X,F8.3,6X,F7.2,8X,F8.2) 756
232   CONTINUE                                                           756
      IF ((HOLDSQ.NE.2.0).AND.(MORE.EQ.2)) GO TO 610                     3MR8CZU
      DO 55 I = 1,NT                                                     756
      RNUM = (TVARCOV(1,1)*TDVMAT(I))-(TVARCOV(I,1)*TDVMAT(1))           756
      RDENOM = SQRT(((TVARCOV(1,1)*TVARCOV(I,I))-TVARCOV(I,1)**2)*((TVAR 756
     1COV(1,1)*SUMSQY)-TDVMAT(1)**2))                                    756
      ZEROR(I,1) = RNUM/RDENOM                                           756
55    CONTINUE                                                           756
      DO 53 I = 2,NT                                                     756
      DO 54 J = 2,I                                                      756
      RNUM = (TVARCOV(1,1)*TVARCOV(I,J))-(TVARCOV(I,1)*TVARCOV(1,J))     756
      RDENOM = SQRT(((TVARCOV(1,1)*TVARCOV(I,I))-TVARCOV(I,1)**2)*((TVAR 756
     1COV(1,1)*TVARCOV(J,J))-TVARCOV(1,J)**2))                           756
      ZEROR(I,J) = RNUM/RDENOM                                           756
54    CONTINUE                                                           756
```

```
 53    CONTINUE                                                       756
       ZEROR(1,1) = 1.0                                               756
       WRITE(6,56)                                                    756
 56    FORMAT (1H0,35HZERO ORDER CORRELATION COEFFICIENTS)            756
       ISTART = -11                                                   756
       IEND = 0                                                       756
       JCOUNT = (NT-1)/12 + 1                                         756
       DO 61 MM = 1,JCOUNT                                            756
       ISTART = ISTART + 12                                           756
       IEND = IEND + 12                                               756
       IFIN = IEND                                                    756
       IF(NT.LT.IEND) IFIN= NT                                        756
       WRITE(6,57) (NAMET(I),I=ISTART,IFIN)                           756
 57    FORMAT((///11X,12(2X,R8)))
       WRITE(6,58)                                                    756
 58    FORMAT(1H0)                                                    756
       DO 60 I = ISTART,NT                                            756
       LNSTOP = I                                                     756
       IF(IFIN.LT.I) LNSTOP = IFIN                                    756
       WRITE(6,59) NAMET(I),(ZEROR(I,J),J=ISTART,LNSTOP)              756
 59    FORMAT(1X,R8,2X,12F10.3)
 60    CONTINUE                                                       756
 61    CONTINUE                                                       756
  610 CONTINUE                                                        OCT18SL
       WRITE(6,320) EXPLSS,EXPLVN,UNEXPSS ,UNEXPVN ,TOTVAR,TOTVN      756
 320   FORMAT (1H0,37X,12HOV VARIATION,14X,11HOV VARIANCE   / 10X,21HEXPL 756
     1AINED BY ALL IV*S,4X,F15.2,10X,F15.2 / 10X,11HUNEXPLAINED,14X,F15. 756
     22,10X,F15.2 / 15X,5HTOTAL, 20X,F15.2,10X,F15.2)                 756
       WRITE(6,300) RSQ,RMULT                                         756
  300 FORMAT(1H0,54H   PROPORTION OF THE VARIATION EXPLAINED BY ALL IVS  756
     1IS,1X,F7.4,10X,14HMULTIPLE R IS ,1X,F7.4)                       756
       WRITE (6,310) NDFONE, NDFTWO, FRATIO                           756
  310 FORMAT (1H0,5X,30HTHE F RATIO TO TEST RSQ (DF*S=,I4,1H,,I4,4H) IS, 756
     12X,F10.4)                                                       756
       IF(HOLDSQ.LE.1.) GO TO 32                                      756
       IF(IVNUM.EQ.2) RETURN                                          756
  311 IF((MORE.EQ.1).OR.(MORE.EQ.2)) GO TO 20                         756
       RETURN                                                         756
       END                                                            756
```

```
      SUBROUTINE  LINEQ      R +-*/                    CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.       PAGE       5

  CARD NO.  SEVERITY          DIAGNOSTIC

        8       1      SDL    PREVIOUSLY DIMENSIONED VARIABLE, FIRST DIMENSIONS WILL BE RETAINED
       97       1             THERE IS NO PATH TO THIS STATEMENT
```

```
        SUBROUTINE  LINEQ      A +-*/                       CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.      PAGE     6

        SYMBOLIC REFERENCE MAP

   ENTRY POINTS
       2  LINEQ

   VARIABLES        SN  TYPE              RELOCATION
    1241  ADDSS         REAL                                     1256  BETAW      REAL       ARRAY
    1231  BTAXY         REAL                                     1337  CITY       REAL       *UNDEF
    1222  DETERM        REAL                                     1230  DFTWO      REAL
       1  DVMAT         REAL          ARRAY     S                1214  EXPLSS     REAL
    1235  EXPLVN        REAL                                     1240  FRATIO     REAL
    1201  GMEANY        REAL                                     1205  HOLDSQ     REAL
    1206  HOLDSS        REAL                                     1210  I          INTEGER
       0  IDEN          REAL          ARRAY      F.P.            1251  IEND       INTEGER
       0  IERR          INTEGER                 ERROR            1254  IFIN       INTEGER
    1250  ISTART        INTEGER                                     0  IVNUM      INTEGER              S
    1211  J             INTEGER                                  1252  JCOUNT     INTEGER
       0  JN            INTEGER                  F.P.            1223  K          INTEGER
    1232  KP            INTEGER                                  1224  L          INTEGER
    1216  LA            INTEGER                                  1217  LB         INTEGER
    1221  LC            INTEGER                                  1220  LD         INTEGER
    1245  LL            INTEGER                                  1255  LNSTOP     INTEGER
    1212  LP            INTEGER                                  1215  LQ         INTEGER
    1253  MM            INTEGER                                  1005  MORE       INTEGER
    1225  MP            INTEGER                                  4635  NAMEPKG    INTEGER    ARRAY     S
    1571  NAMET         INTEGER       ARRAY                        63  NCASE      INTEGER              BETA
    1226  NOFONE        INTEGER                                  1227  NOFTWO     INTEGER
      62  NEVNT         INTEGER                 BETA             4623  NIVPKG     INTEGER              S
    1207  NT            INTEGER                                    61  NTIME      INTEGER              BETA
    1200  NUMCASE       INTEGER                                  1242  PARTNUM    REAL
    1244  PARTR         REAL                                     1243  PARTSQ     REAL
       0  PKGNO         INTEGER                  F.P.            1247  RDENOM     REAL
    1234  RMULT         REAL                                     1246  RNUM       REAL
    1213  RSQ           REAL                                     2014  SIGN       REAL       ARRAY
       0  SOL           REAL          ARRAY     BETA             1346  STDERR     REAL       ARRAY
    1733  STDEV         REAL          ARRAY                      4624  SUMSQY     REAL                 S
    1510  TDVMAT        REAL          ARRAY                      4625  TITLE      REAL       ARRAY     S
    1202  TOTVAR        REAL                                     1203  TOTVN      REAL
    1427  TTEST         REAL          ARRAY                         0  TVARCOV    REAL       ARRAY      F.P.
    1233  UNEXPSS       REAL                                     1236  UNEXPVN    REAL
      62  VARCOV        REAL          ARRAY     S                   0  VERT       REAL       ARRAY      F.P.
    1652  XMEAN         REAL          ARRAY                      1237  XNIVPKG    REAL
    1204  YSTDEV        REAL                                        0  ZEROR      REAL       ARRAY      F.P.

   FILE NAMES         MODE
        TAPE6          FMT

   EXTERNALS          TYPE    ARGS
        INVERT                  6                                       SQRT       REAL       1 LIBRARY

   STATEMENT LABELS
       0  10                                   0  11                                          65  20
       0  21                                   0  22                                           0  23
       0  24                                   0  25                                           0  26
```

```
STATEMENT LABELS
     0  31          INACTIVE              176  32                                  175  33
     0  53                                  0  54                                    0  55
  1106  56      FMT                      1114  57      FMT                        1117  58      FMT
  1121  59      FMT                         0  60                                    0  61
     0  70                                202  99                                    0  100
     0  110                                 0  120                                   0  130
     0  140                               432  159                                1050  160     FMT
  1065  230     FMT                      1073  231     FMT                           0  232
  1077  233     FMT                         0  248                                 337  249
     0  250                              1144  300     FMT                        1157  310     FMT
     0  311         INACTIVE             1037  313     FMT                        1033  314     FMT
  1124  320     FMT                      1006  510     FMT                         714  610
     0  700                                 0  1141                                274  1142
     0  1145        INACTIVE              271  1146                                301  1150
  1012  1160    FMT                         0  1210                               1030  1220    FMT
   327  3000

COMMON BLOCKS   LENGTH
       ERROR         1
       S          2510
       BETA         52

STATISTICS
  PROGRAM LENGTH    2156B     1134
   COMMON LENGTH    5003B     2563
```

```
      SUBROUTINE INVERT (A,N,B,M,DETERM,M6  )                           756
C     SUBROUTINE MATINV(A,N,B,M,DETERM)                                 756
C                                                                       756
C     MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS   756
C                                                                       756
C     SUBROUTINE MATINV.                                                756
C                                                                       756
C     PROGRAMMED BY BURTON S. GARBOW, ARGONNE NATIONAL LABORATORY,      756
C     AND REPORTED IN IBM 704-709 SHARE LIBRARY AS AN F402.             756
C     THIS SUBROUTINE COMPUTES THE INVERSE AND DETERMINANT OF           756
C     MATRIX A, OF ORDER N, BY THE GAUSS-JORDAN METHOD.  A-INVERSE      756
C     REPLACES A, AND THE DETERMINANT OF A IS PLACED IN DETERM.  IF     756
C     M = 1 THE VECTOR B CONTAINS THE CONSTANT VECTOR WHEN MATINV IS    756
C     CALLED, AND THIS IS REPLACED WITH THE SOLUTION VECTOR.  IF M = 0, 756
C     NO SIMULTANEOUS EQUATION SOLUTIONS ARE CALLED FOR, AND B IS NOT   756
C     PERTINENT, N IS NOT TO EXCEED 50.                                 756
C     A, N, B, M, AND DETERM IN THE ARGUMENT LIST ARE DUMMY VARIABLES.  756
C                                                                       756
C                                                                       756
      DIMENSION IPIVOT(50), A(M6,M6), B(M6,1), INDEX(50,2), PIVOT(50)   756
      EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)         756
C     INITIALIZATION                                                    756
  100 DETERM=1.0                                                        756
  110 DO 120 J=1,N                                                      756
  120 IPIVOT(J)=0                                                       756
  130 DO 560 I=1,N                                                      756
C     SEARCH FOR PIVOT ELEMENT                                          756
  140 AMAX=0.0                                                          756
  150 DO 240 J=1,N                                                      756
  160 IF (IPIVOT(J)-1) 170, 240, 170                                    756
  170 DO 230 K=1,N                                                      756
  180 IF (IPIVOT(K)-1) 190, 230, 680                                    756
  190 IF (ABS (AMAX)-ABS (A(J,K))) 200, 230, 230                        756
  200 IROW=J                                                            756
  210 ICOLUM=K                                                          756
  220 AMAX=A(J,K)                                                       756
  230 CONTINUE                                                          756
  240 CONTINUE                                                          756
  250 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                                   756
C     INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL                 756
  260 IF (IROW-ICOLUM) 270, 370, 270                                    756
  270 DETERM=-DETERM                                                    756
  280 DO 310 L=1,N                                                      756
  290 SWAP=A(IROW,L)                                                    756
  300 A(IROW,L)=A(ICOLUM,L)                                             756
  310 A(ICOLUM,L)=SWAP                                                  756
  320 IF(M) 370, 370, 330                                               756
  330 DO 360 L=1, M                                                     756
  340 SWAP=B(IROW,L)                                                    756
  350 B(IROW,L)=B(ICOLUM,L)                                             756
  360 B(ICOLUM,L)=SWAP                                                  756
  370 INDEX(I,1)=IROW                                                   756
  380 INDEX(I,2)=ICOLUM                                                 756
  390 PIVOT(I)=A(ICOLUM,ICOLUM)                                         756
  400 DETERM=DETERM*PIVOT(I)                                            756
```

```
      C      DIVIDE PIVOT ROW BY PIVOT ELEMENT                           756
        410 A(ICOLUM,ICOLUM)=1.0                                          756
        420 DO 430 L=1,N                                                  756
        430 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)                              756
        440 IF(M) 470, 470, 450                                           756
        450 DO 460 L=1,M                                                  756
        460 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)                              756
      C      REDUCE NON-PIVOT ROWS                                        756
        470 DO 560 L1=1,N                                                 756
        480 IF(L1-ICOLUM) 490, 560, 490                                   756
        490 T=A(L1,ICOLUM)                                                756
        500 A(L1,ICOLUM)=0.0                                              756
        510 DO 520 L=1,N                                                  756
        520 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T                                 756
        530 IF(M) 560, 560, 540                                           756
        540 DO 550 L=1,M                                                  756
        550 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T                                 756
        560 CONTINUE                                                      756
      C      INTERCHANGE COLUMNS                                          756
        570 DO 670 I=1,N                                                  756
        580 L=N+1-I                                                       756
        590 IF (INDEX(L,1)-INDEX(L,2)) 600, 670, 600                      756
        600 JROW=INDEX(L,1)                                               756
        610 JCOLUM=INDEX(L,2)                                             756
        620 DO 660 K=1,N                                                  756
        630 SWAP=A(K,JROW)                                                756
        640 A(K,JROW)=A(K,JCOLUM)                                         756
        650 A(K,JCOLUM)=SWAP                                              756
        660 CONTINUE                                                      756
        670 CONTINUE                                                      756
        680 RETURN                                                        756
            END
        SUBROUTINE  INVERT    R +-*/                                 CDC 6600 FTN V3.0-V359 OPT=1  75/06/13. 05.55.11.      PAGE      3

        SYMBOLIC REFERENCE MAP

  ENTRY POINTS
      2  INVERT

  VARIABLES      SN  TYPE          RELOCATION
      0  A           REAL        ARRAY     F.P.            255  AMAX      REAL
      0  B           REAL        ARRAY     F.P.              0  DETERM    REAL                        F.P.
    247  I           INTEGER                               254  ICOLUM    INTEGER
    340  INDEX       INTEGER     ARRAY                     256  IPIVOT    INTEGER     ARRAY
    253  IROW        INTEGER                               246  J         INTEGER
    254  JCOLUM      INTEGER                               253  JROW      INTEGER
    250  K           INTEGER                               251  L         INTEGER
    252  L1          INTEGER                                 0  M         INTEGER                     F.P.
      0  M6          INTEGER               F.P.              0  N         INTEGER                     F.P.
   2372  PIVOT       REAL        ARRAY                     255  SWAP      REAL
    255  T           REAL

  INLINE FUNCTIONS    TYPE    ARGS
         ABS          REAL      1  INTRIN

  STATEMENT LABELS
      0  100          INACTIVE               0  110          INACTIVE                0  120          
      0  130          INACTIVE               0  140          INACTIVF                0  150          INACTIVE
      0  160          INACTIVE               0  170          INACTIVE                0  180          INACTIVE
      0  190          INACTIVE               0  200          INACTIVE                0  210          INACTIVE
      0  220          INACTIVE              54  230                                 57  240
      0  250          INACTIVE               0  260          INACTIVE                0  270          INACTIVE
      0  280          INACTIVE               0  290          INACTIVE                0  300          INACTIVE
      0  310                                 0  320          INACTIVE                0  330          INACTIVE
      0  340          INACTIVE               0  350          INACTIVE                0  360
    112  370                                 0  380          INACTIVE                0  390          INACTIVE
      0  400          INACTIVE               0  410          INACTIVE                0  420          INACTIVE
      0  430                                 0  440          INACTIVE                0  450          INACTIVE
      0  460                               145  470                                  0  480          INACTIVE
      0  490          INACTIVE               0  500          INACTIVE                0  510          INACTIVE
      0  520                                 0  530          INACTIVE                0  540          INACTIVE
      0  550                               202  560                                  0  570          INACTIVE
      0  580          INACTIVE               0  590          INACTIVE                0  600          INACTIVE
      0  610          INACTIVE               0  620          INACTIVE                0  630          INACTIVE
      0  640          INACTIVE               0  650          INACTIVE                0  660
    233  670                               236  680

  STATISTICS
    PROGRAM LENGTH    2534B     1372
```

BIBLIOGRAPHICAL REFERENCES

Aldrich, D. G., III. An Analysis of Vocational Program Costs. 1972. (ED 075 597).

American Institute for Research. Career Education: An Annotated Bibliography for Teachers and Curriculum Developers. U.S. Office of Education, 1973.

_______. Career Education: A Curriculum Design and Industrial Objectives Catalogue. U.S. Office of Education, 1973.

Baldwin, L. P. A Cost Accounting Model to Assess Actual Cost of Vocational and Nonvocational Courses. Office of the Chancellor, California Community Colleges, July, 1970.

Bailey, L. J. Facilitating Career Development: An Annotated Bilbiography. 1970. (ED 034 062).

Bailey, L. J., and R. Stadt. Career Education: New Approaches to Human Development. Bloomington, Illinois: McKnight Publishing Company, 1973.

Barsky, S. L. The Application of Cost-Benefit Analysis in Manpower Areas. 1970. (ED 069 890).

Bloom, B. S., Hastings, J. T., and G. F. Madaus. Handbook on Formative and and Summative Evaluation of Student Learning. New York: McGraw-Hill, 1971.

Bloom, B. S. (ed.) Taxonomy of Educational Objectives. Handbook I: Cognitive Domain. New York: David McKay, 1956.

Cox, J. A. Utah Project "Follow-Up": Interim Report. 1969. (ED 033 236).

Borus, M. E., and W. R. Tash. Measuring the Impact of Manpower Programs, A Primer. Ann Arbor, Michigan: Institute of Labor and Industrial Relations, The University of Michigan--Wayne State University.

Division of Comprehensive and Vocational Education Research. Four Years of Research, Development, and Training: A Bibliography; July, 1964 to June, 1968. (ED 075 597).

Dorfman, Robert. Measuring Benefits of Government Investments. Washington, D. C.: The Brookings Institution, 1963.

Economic Council of Canada: 8th Review. Design for Decision-Making: An Application to Human Resources Policies. Ottawa, Canada: Queen's Printers, 1971.

Goldhammer, K. A Career Currcilum. Columbus, Ohio: The Center for Vocational and Technical Education. The Ohio State University, December, 1971.

Harris, G. W. and O. K. O'Fallon. Cost Analysis of Secondary School Vocational Technical Education Programs. Tennessee State Board for Vocational Education. 1973. (ED 080 897).

Hu, T. W., Lee, M. L., and E. W. Stromsdorfer. A Cost-Effectiveness Study of Vocational Education. Final Report Project No. O. E. 512, U.S. Department of Health, Education, and Welfare, Office of Education, March, 1969.

Koch, J. V. A Benefit-Cost Analysis of Vocational-Occupational Training at Selected Illinois Junior Colleges. 1972. (ED 033 236).

Krathwohl, D. R., Bloom, B. S., and B. B. Masis. Taxonomy of Educational Objectives. Handbook II: Affective Domain. New York: David McKay, 1964.

Little, J. K. Review and Synthesis of Research on the Placement and Follow-Up of Vocational Education Students. 1970. (ED 040 274).

Marland, Sidney P., jr. Career Education Now. Presentation at the National Association of Secondary School Principals in Houston, Texas. January 23, 1971. (ED 048 480).

_______. Career Education--More Than a Name. Presentation at the Annual Meeting of the State Directors of Vocational Education in Washington, D.C. May 4, 1971. (ED 050 295).

_______. Educating for the Real World. Presentation to the Twelfth Annual Banquet of the Jefferson County Chamber of Commerce in Harpers Ferry, West Virginia. May 26, 1971.

McClure, Larry, and Carolyn Buan (eds.) Essays on Career Education. U.S. Government Printing Office, Washington, D.C. April, 1973.

McCracken, David. Work Experience for Broadening Occupational Offerings: A Selected Bibliography for Use in Program Development. 1969. (ED 034 062).

Moayed-Dadkhah, K., Stromsdorfer, E. W., and H. Fishkind. Cost Analysis of Manpower Programs: An Analysis of the State of the Art. U.S. Department of Labor Contract No. P . O. 73-3814, Bloomington, Indiana: Department of Economics, Indiana University, November, 1973.

Ohio State University. A Comprehensive Career Education Model. Columbus, Ohio: The Center for Vocational and Technical Education. (undated).

Olivo, C. T. Research Visibility: Vocational Education Is People With Needs Beyond the High School. 1967. (ED 016 871).

Puerto Rico State Department of Education. A Cost Effectiveness Analysis of Vocational Education Programs in Puerto Rico. New York: Peat, Marwick, Mitchell and Co., 1971. (ED 073 303).

Roney, M. W. and P. V. Braden. Occupational Education Beyond the High School in Oklahoma; An Analytical Study with Recommendations for a Statewide System of Manpower Development. Oklahoma State University, 1968. (ED 020 395).

Schill, W. J. Concurrent Work-Education Programs in 50 States: 1965-66. (ED -23 886). W

Somers, G. G., and W. D. Woods. (eds.). Cost-Benefit Analysis of Manpower Policies. Proceedings of a North American Conference. Kingston, Ontario: Industrial Relations Centre, Queen's University at Kingston. At Wisconsin: Madison, Wisconsin, University of Wisconsin, Center for Studies in Vocational and Technical Education. 1969.

Somers, G. G., and E. W. Stromsdorfer. A Cost-Effectiveness Study of the In-School and Summer Neighbourhood Youth Corps. Madison, Wisconsin: Industrial Relations Research Institute, Center for Studies in Vocational and Technical Education. The University of Wisconsin, 1970.

Stromsdorfer, E. W. Review and Synthesis of Cost Effectiveness Studies of Vocational and Technical Education. Columbus, Ohio: ERIC Clearinghouse on Vocational and Technical Education. Ohio State University, 1972.

U.S. Office of Education Abstract. School-Based Comprehensive Career Education Model. July, 1971.

Wallace, H. R. Review and Synthesis of Research on Cooperative Vocational Education. 1970. (ED 040 274).